AN ARGUS SPECIALIST PUBLICATION

February/March 1985

ZX COMPUTING

£1.95

Britain's Best Buy For The Sinclair User

Over 130 pages of information and programs for the ZX Spectrum, ZX81 and ZX80

Attention super-sleuths – can YOU solve the mystery of OODUNNIT?

Defend our shores from the Beach Attack

Take your woods and irons for a day at Netherwallop Golf course

Battle of the giants – Cyrus IS chess and Superchess 3.0 in a fight to the finish

Plus all our usual news, reviews and features




COMBAT LYNX
From DURELL
(Technical Support from Westland Helicopters)
Available now from most retail outlets
COMMODORE 64
SPECTRUM 48k
Available soon
ORIC – BBC
AMSTRAD
DURELL sales dept.,
Castle Lodge, Castle Green, Taunton, Somerset, TA1 4AB
TIM HAYWARD

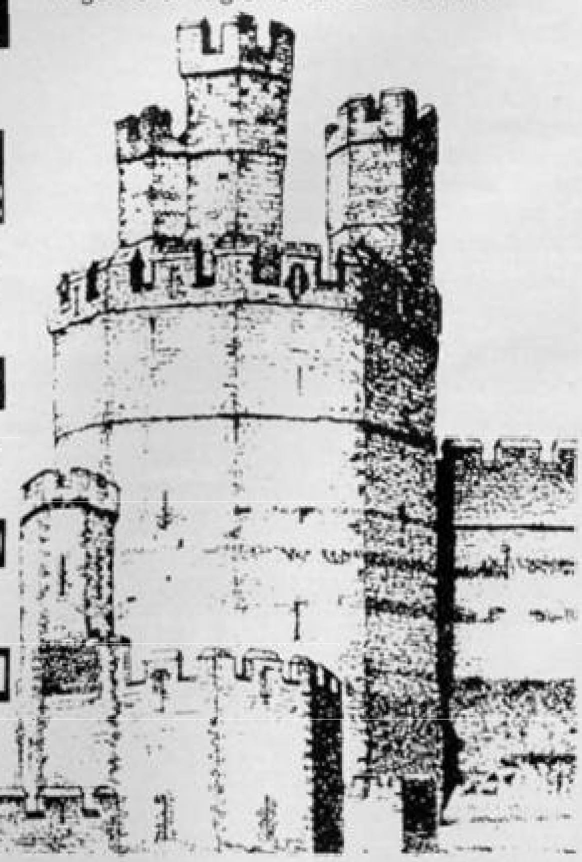
ZX COM

Editor: Ray Elder
Editorial Assistant: Jamie Clary
Group Editor: Wendy J Palmer
Sales Executive: Jonathan McGary
Advertisement Manager: Barry Bingham
Divisional Advertising Manager: Chris Northam
Copy Controller: Sue Couchman
Chairman: T J Connell

Origination and design by MM Design & Print, Circus House, 26 Little Portland Street, London W1N 5AF.

Published by Argus Specialist Publications Ltd, 1 Golden Square, London W1R 3AB.

CONTENTS



ASP FIGHTS SOFTWARE PIRACY

Much has been said and written in condemnation of software piracy but few have taken a positive stand against it. ASP is among those few that have taken action to help curb the grave problem of home copying of commercial software.

ASP has already taken steps to eliminate advertisements in our magazines which relate to tape duplication for piracy purposes. While it is appreciated that individuals may take 'back-up' copies of their own programs, it should be noted that it is *ILLEGAL* to copy commercially available software for other than personal use.

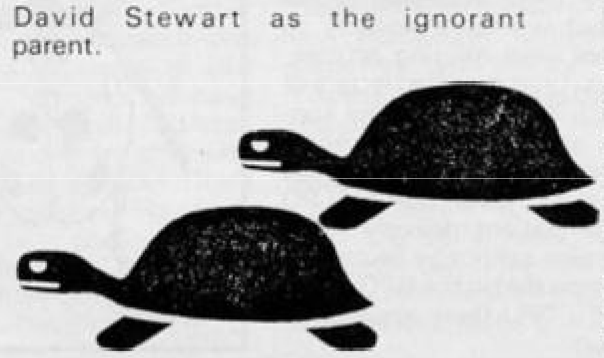
Software piracy is costing the software industry huge sums of money which is detrimental to the future development of the industry. It is in everybody's interests to dramatically reduce the level of software piracy primarily because firms need funds raised from software sales to plough back into research and development of new products. This means that the standard of software products can only improve.

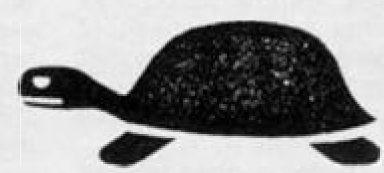
ASP hopes our action will help combat this serious problem in order to maintain and improve the high standards of the UK software industry. We are asking *you* to do the same by refraining from duplicating or copying commercially available software for anything other than personal use.

MPUTING

ZX Computing is published bi-monthly on the fourth Friday of the month. Distributed by: Argus Press Sales & Distribution Ltd, 12-18 Paul Street, London EC2A 4JS. 01-247 8233. Printed in the UK by: Garnett Print, Rotherham and London.







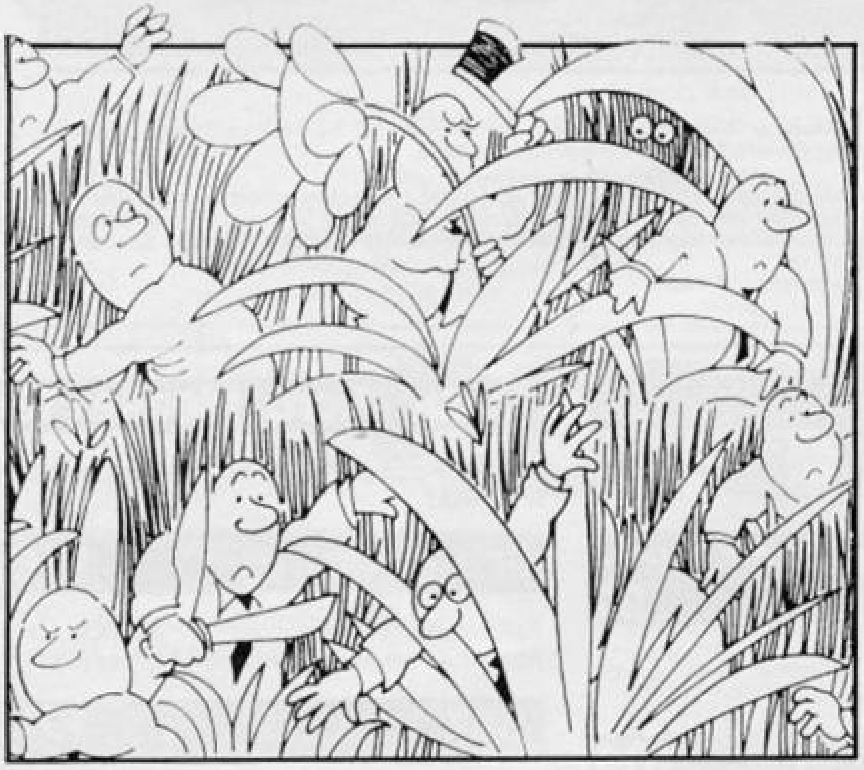
Subscription rates: UK £12.70 including postage. Airmail and other rates upon application to ZX Computing, Subscriptions, Infonet Ltd, Times House, 179 The Marlowes, Hemel Hempstead, Herts HP1 1BB (Tel: 0442 48432)

ZX Computing is constantly on the look-out for well-written articles and programs. If you think that your efforts meet our standards, please feel free to submit your work to us for consideration for publication.

All submitted material should be typed if possible; handwritten work will be considered, but please use your neatest handwriting. Any programs submitted should be listed, a cassette of your program alone will not be considered. All programs must come complete with a full explanation of the operation and, where relevant, the structure; Spectrum programs should be accompanied with a cassette of the program as well as the listing.

All submissions will be acknowledged and the copyright in such works which will pass to Argus Specialist Publications Ltd will be paid for at competitive rates. All work for consideration should be sent to the Editor at our Golden Square address.

WELCOME



I have an axe to grind.

I don't usually like to grind axes in public, because, as those of you who have indulged in this activity know, the sparks have a tendancy to fly and the grinder can easily get burnt.

But still, the time has come as the walrus said, to speak of many things, and one thing that has increasingly irritated me is the service (?) provided by the GPO.

Now people in glass houses shouldn't, as they say, and I am certainly not in a position to claim that I'm fault free. Goodness Knows there are errors enough in *ZX Computing* and it drives me to drink trying to eliminate them, the occasional spelling mistake, lapse of grammar and printing problem. But I am aware of them and I try to keep them to a minimum by aiming for perfection.

The excellent DK'Tronics club newsletter arrived the other day and my comment about the unreliability of the GPO was quoted in it. It made me take stock and I was shocked at how much has gone astray.

I know for certain that in the last few months four items from DK, TWO Spectrums from Sinclair, Software from Melbourne House, Automata, Micromega, Ultimate and Beyond has been sent and never reached us.

In the other direction review packages for two of our reviewers never reached them. This meant that Chinese Juggler, Beach Head, Automania, TLL, The Prince, Alien and several others didn't get reviewed, the companies concerned would willingly supply us with replacements, and many have done so, but the time delay in this fast turnover market often means that it is too late.

The latest example is the Adapt RGB interface, I have been waiting for this device for some six weeks now, and I had planned to include in this issue an article on this unit and some of the monitors on the market. Opus sent theirs along five weeks ago and several others are sitting here waiting, taking up room and time as I have to explain to the companies that so kindly loaned them to us that we are "not quite finished with it yet"

Even more worrying is the fact that we have no way to know of your missing articles, reviews, programs and letters of praise (!) Therefore, to the suppliers and to yourselves, I apologise. I would also add that at the public level, our local PO and the patient delivery men, the service cannot be faulted. It just seems that in the GPO's version of a CPU there appears to be a bug.

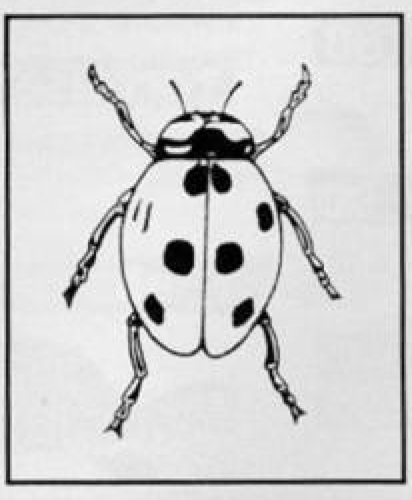
Contributions

We are always on the lookout for good programs and articles for future issues of *ZX Computing,* and where better to look than to our own readers. If, when reading through the magazine you think you can write programs as well, or better than, our present contributors, then let's hear from you.

All contributions are, of course, paid for at very competitive rates. So if you've got your eye on a new ZX add-on or you'd just like to supplement your pocket money, get writing! It is vital, though, that all the programs you send us are totally original, and not 'borrowed' or 'adapted' from other magazines or books. (When Tim Hartnell was sitting in the Editor's chair, he even recieved 'original' contributions he himself had written for his own books!)

Any kind of program (business, domestic, educational, or just fun) will be welcomed which use ZX BASIC in clever and efficient ways, or those which employ certain routines which can be re-used on other programs.

Program listings are vital, along with a clear explanation of how the program is constructed, what it does and what the user can expect to see once the program is RUN (a screen dump is particularly valuable in this respect). When submitting Spectrum programs, it is very important to remember to enclose a cassette of the program as well as the listing, as this will allow us to check the program before publication.



On The Other Hand

BT's Prestel/Micronet 800 seems to be getting a fair bit of use, several readers have Mailboxed messages to me and most comments were favourable, although some criticism of the breakdowns was made.

There are a few people whom I failed to reply because I pressed the wrong button at the wrong time and lost their numbers, however most were answered.



Wade Biss suggested we ran a regular column in *ZX Computing*, this is a good idea, but I think I ought to get someone with a bit more expertise than myself to write it! I am making enquiries. . . .

Meanwhile there is a superb book published by Melbourne House on Micronet 800 and the Spectrum and the VTX 5000 by Alan Giles, well worth looking at.

Special thanks for M/Boxes sent by Mr. A. La Delfa, Mrs. C.V. Russel, N. Pearce of Weston, Anne Younger, and whoever applied to Meditel on my behalf!

Introducing . . .

The last issue featured Spectrum arcade reviews by David Howard, he is a well known author and his Interface book on arcade games for the Spectrum is a steady seller.

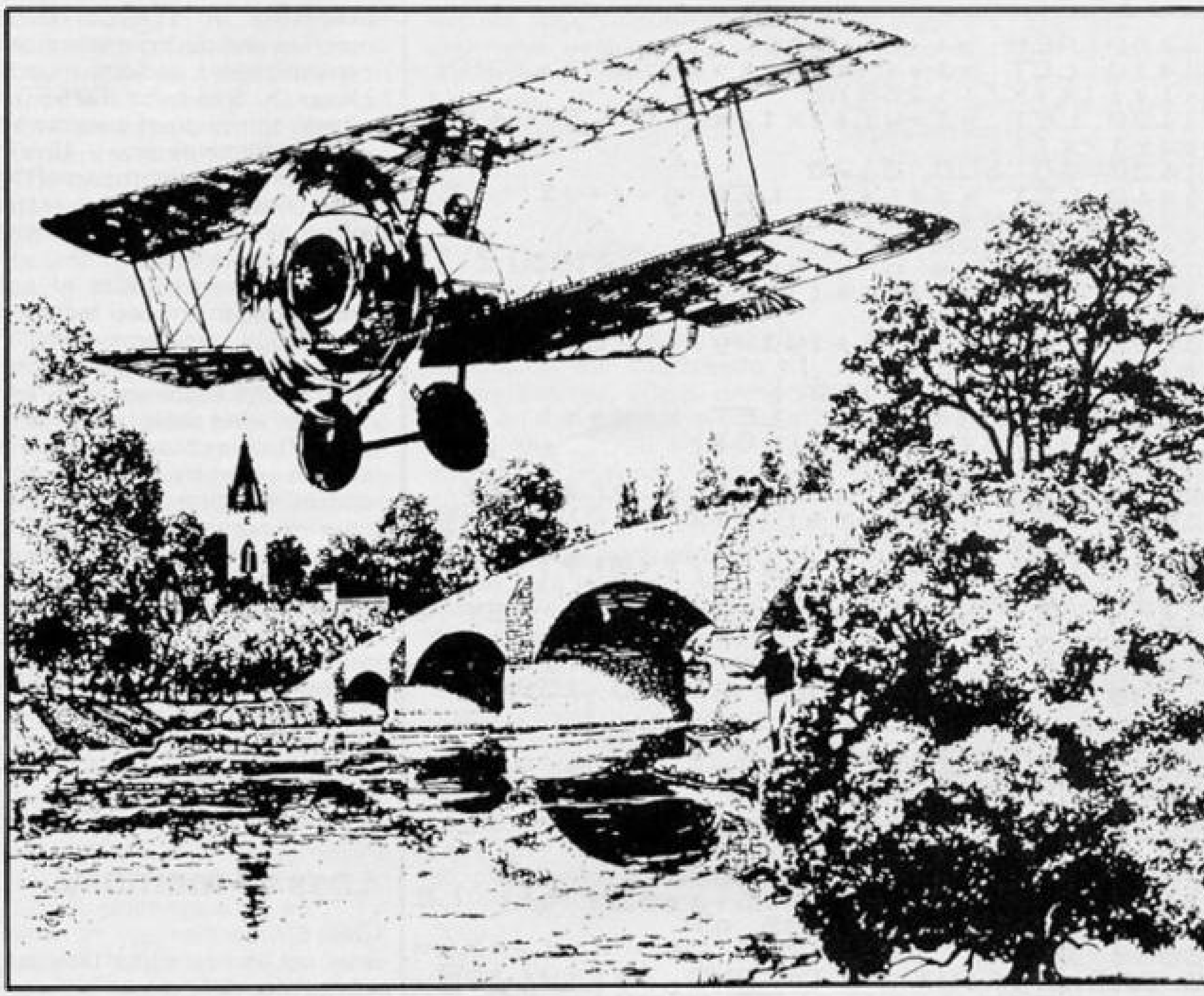
He has agreed to join us on a regular basis as chief arcade program review, we hope you enjoy his assessment of these programs and look forward to a lengthy association.

And so here we go again, apologise to all those with problems with our Christmas program (and to Clyde Bish who wrote it), we sent out amendments to everyone who enquired about it as it would have been too late just to print them in this issue.

I just hope the GPO didn't lose any. . .

If you do find a problem then, after checking to make sure it's not your error, please do write to use, we always do our best to help.

Inside is another collection of some of the best software published, critical comment and informative articles (modest aren't I) so chocks away Biggles, and away we go. . . .



Life in it yet . . .

Dear *ZX Computing*,
Having read Mr. Rogers letter in the August/September since about there still being life left in the ZX81, I would like to add the following comments on the value of the ZX81. It is easy for experienced users of computers to be cynical towards it because of its limitations. As they have had several years of computing they are quite often unable to appreciate it's value. It is an ideal computer for a beginner. It being cheap it is not such a waste when it is discarded after a month or two maybe. It is easy to learn on and takes a novice from the first steps of programming to be a competent user. It is easier to start at the beginning rather than trying to get complicated and not really understanding it properly. The simple ZX81 machine can be built up progressively with useful hardware and interesting software to make a worthwhile system of equipment. Or also it could be sold to be put towards a bigger machine.

The ZX81 is still the cheapest computer on the market and no other company has produced another computer on par for value with it.

The ZX81 will still provide enjoyment for many people for a while to come.

Yours faithfully
A.R. PARKINS.

QL Microdrives

Dear Sir,
Sinclair Research have been much criticised in the past for launching products before they are actually available, but I am pleased to say that this trend now seems to have been reversed. I seem to have found a product which is readily available *before* Sinclair have launched it – extra Microdrives for the QL!

On the right-hand edge of the QL, near the reset button is the slot designated for Microdrives to augment the two built-in to the QL. Page 2 of the Introduction to my QL Users' Guide states "ZX Microdrives are not suitable for use with the QL", but recently I have read a letter in the computing press (sorry, forgotten where), that Spectrum Microdrives could be used as read-only devices with the QL. Filled with trepidation, I was not only able to confirm this, but was able to FORMAT and SAVE files on cartridges in these extra drives. Such files are LOADable in the two original drives, and I have now been using the QL with two extra ZX Microdrives for a week with no apparent snags. I'm glad I made provision for eight microdrives in my QL Utilities package!

To connect them, I unplugged the ribbon cable from the Spectrum's Interface 1, and plugged this into the Microdrive expansion port of the QL. The Microdrives therefore end up with the *slots facing away from you.* Perhaps this is why Sinclair say they are not suitable! Is this compatibility exclusive to QLs fitted with "dongles" (I have QDOS version PM till)? I do hope some of you will try this with later QLs and let us know. Don't sue me if something goes wrong; the User Guide must be right sometimes!

Yours faithfully
C.C. Wilton-Davies)

Slogo 'no-go!'

Dear Ray,
I trust that you may find the following comments sufficiently interesting for publication although by this time no doubt you will have received other letters on the same subject.

Having entered David Nowotnik's "Slogo" Program the other night I set about the usual trial running exercise in order to eliminate the inevitable entry errors. However after this process appeared complete I found that I still had a bug in the "X wrap" routine which only appeared when the direction of travel was between RT 90 and RT 180. I eventually traced this to line 1420 which is patently wrong (yl should be equal to – x-l/TAN(PI-q). However there is an even more serious flaw which occurs in line 1425 with the command RT 90 FD 200 (or any distance off the screen) which indicates clearly the risks in using the reciprocal of a function which can by itself have values of 0 and infinity.
The use of TAN q is of course not required (nor for that matter is Pythagoras) as anyone familiar with congruent triangles will realise.

If the variables xf and yf represent the increments in the x and y directions corresponding to a distance along the direction of travel a then the final coordinates will be (xf + x),(yf + y). If this point is within the screen limits the whole distance a can be plotted. If one of these coordinates lies outside the limits of the screen then only a proportion, say al, of the total distance can be plotted within the screen limits. If say the x co-ordinate for instance is exceeded then it is easy to calculate the distance along the x direction from the point x,y to the edge of the screen. (Either 255-x or x). If this distance is designated xl then congruency tells us that al/a = xf. (Similar considerations also apply when it is the y screen limit that is exceeded.

I enclose amended "wrap" and "enter" routines which are based on the above and which are rather neater than those published (as well as working in all directions). The routines at lines 1400 and 1500 merely calculate the length of line in the direction of travel that will fit into the screen, draws it, plots the point at the start of the "wrap" and draws the remainder of the line assuming that this will now fit; if it doesn't then the process is repeated till it does. In the case where both the final coordinates lie outside the screen limits then the routine at line 1600 determines the distances along the line of travel for the drawn line to exceed the x axis ▶

```
1400 REM ** x wrap **
1410 LET x1=(-x*((x+xf)<0)+(255-
x)*((x+xf)>255))
1420 LET y1=yf*(x1/xf): LET a1=a
*(x1/xf)
1430 GO SUB 3120
1440 LET a=a-a1: LET y=y+y1: LET
 x=(0+255*((xf+x)<0))
1450 RETURN
1500 REM ** y wrap **
1510 LET y1=-y*((y+yf)<0)+(175-y
)*((y+yf)>175)
1520 LET x1=xf*(y1/yf): LET a1=a
*(y1/yf)
1530 GO SUB 3120
1540 LET a=a-a1: LET x=x+x1: LET
 y=(0+175*((yf+y)<0))
1550 RETURN
1600 REM ** x and/or y wrap **
1610 LET x1=(-x*((x+xf)<0)+(255-
x)*((x+xf)>255))
1620 LET y1=-y*((y+yf)<0)+((175-
y)*((y+yf)>175))
1630 LET ay=a*ABS (y1/yf): LET a
x=a*ABS (x1/xf)
1640 IF ay>=ax THEN GO TO 1400
1650 IF ax>ay THEN GO TO 1500
```

```
3000 REM ** forward **
3020 GO SUB 1200: IF err>0 THEN
RETURN
3030 IF turt=0 THEN LET turt=1:
GO SUB 1100
3040 LET q=dir*PI/180
3050 LET xf=INT (.5+a*SIN q): LE
T yf=INT (.5+a*COS q)
3060 IF (xf+x)<=255 AND (xf+x)>=
0 AND (yf+y)<=175 AND (yf+y)>=0
THEN LET x1=xf: LET y1=yf: GO SU
B 3120: LET x=x+x1: LET y=y+y1:
RETURN
3070 IF ((xf+x)>255 OR (xf+x)<0)
 AND (yf+y)<=175 AND (yf+y)>=0 T
HEN GO SUB 1400
3080 IF (xf+x)<=255 AND (xf+x)>=
0 AND ((yf+y)>175 OR (yf+y)<0) T
HEN GO SUB 1500
3090 IF ((xf+x)>255 OR (xf+x)<0)
 AND ((yf+y)>175 OR (yf+y)<0) TH
EN GO SUB 1600
3100 GO TO 3050
3120 IF pp=1 THEN PLOT x,y: DRAW
 x1,y1:
3130 IF pp=0 THEN PLOT INVERSE 1
; OVER 1;x,y: DRAW INVERSE 1; OV
ER 1;x1,y1
3140 LET t=0: RETURN
```

and y axis limits, compares them and then routes the program to the routine for the appropriate axis reached first.

Eureka . . . and Euclid knew a thing or too as well!

David F M Carmichael

A Tasword tale

Dear Sir,

Tasword Two is a most excellent word processing program for the Spectrum and to do it justice a suitable printer is a necessity. As a complete ignoramus, a good deal of assistance in making a choice was needed. However, after all the dealers that I could discover in Cumbria had been visited or telephone to enquire a) whether they had printers in stock or b) whether they could make available examples of the work produced it was clear that help was not readily available. Only one had a printer in stock while the others were not interested or did not keep a promise to get in-to touch.

The next step was to consult as many periodicals as I could find for reviews of the various printers on the market. Reviews turned out to be very few but it seemed that Seikosha and Microline offered possibilities and so I wrote to both DRG Business Systems and the agents for Microline. Within three days DRG had sent literature on the Seikosha machine and during a telephone conversation I decided to purchase a Seikosha GP550A which, to my great surprise arrived within three days — I live in a beautiful but remote part of the Lake District. Two weeks later information on the Microline printers came to hand.

Over seventy years of age and lacking in any real technical knowledge, problems arose in interpretation of instructions. Both of the suppliers, DRG and Tasman were most helpful with my difficulties but the first class service given by DRG is worth special mention. When a supplier gives such ungrudging after-sales halp the world should be told. Management of Hilton Computer services have also been most tolerant in answering my idiotic questions.

Yours faithfully

Alex Ames

High Lorton

Cumbria.

Loss of memory

Dear Sir,

I've asked you some time ago about the differences between the differents issues of Spectrum (1,2,3 and 3B), and you told me that, "To my knowledge, *and Sinclair confirm,* ALL hardware is compatible with no problems for all issues."

Unfortunately, I have discovered a big difference between the issue 3 (at least some issue 3) and the others . . . Let me tell you my history . . . . . .

In May, my Spectrum crashed, and, due to the problems to repair it in Spain (it was imported from UK), I decided to repair it myself. I opened it, and I found some problem on the first 16K RAM storage.

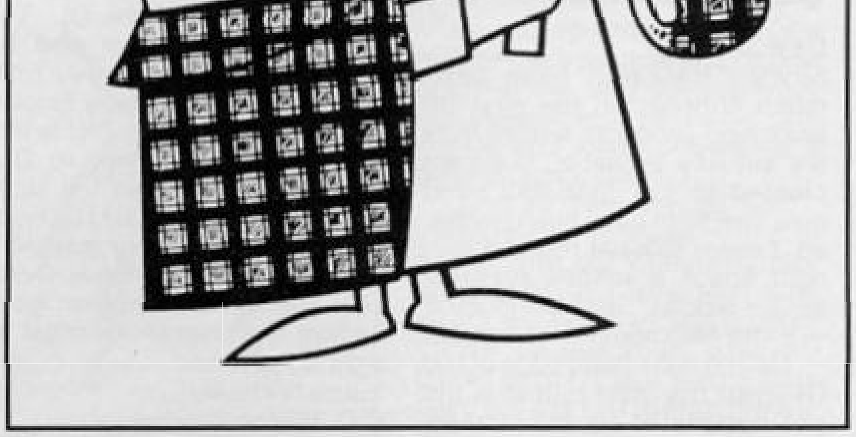
Now start the problems. My Spectrum has 16K MM5290N-2 circuits, and not the standard 4116-2. I had a look to the I.C. Master Book, and the 4116-2 appears as a valid replacement for the MM5290N-2, *BUT*:

If I insert the 4116-2 in the place of the MM5290N-2, the Spectrum does not initialize and in the TV screen there are random dynamic squares with random colours.

I forced a permanent RESET signal, in order to have some stable waveforms in the data bus, and, using an oscilloscope at the data pin of the I.C., I found the data bit missing in some moments, and I feel there cold be some "refresh" problems. In fact, the "REFRESH" signal of the -80, used during the TV Sync. cycle, is not connected in the issue 3. Maybe the MM5290N-2 accepts this "non connexion and non refreshing cycle" due to a longer refresh time, and the 4116-2 does not?

As a last test, in order to check if there was any problem in any other I.C. connected to the same bit of the data bus, I interchanged a "good" MM5290N-2 with a 4116-2, and the above waveforms went with the I.C. to the new places, so there is not a problem with others I.C.'s, it is a problem related with the 4116-2 itself.

An my history does not end yet. I've tried to get MM5290N-2 in Spain, UK, Germany . . . impossible. The best answer I've got is "it is an ob-



soleted circuit'' and it seems that all the circuits that there is at the moment in the world are in Sinclair's stock, and they do not sell them . . . . .

Some friend of mine is trying to get them all, over the world. I'll let you know if I arise to get them . . . . . .

By the way, it is not easy for me to send my Spectrum to Sinclair UK to repair it (even waiting some 6 months to get it back) and the Sinclair repres. in Spain does not repair it because it was not bought in Spain.

I've read in some of your last issues you were looking for contributions to publish in your magazine. I do not know if this one will fit in your ideas of what could be published or not in your magazine, but I feel that, with some corrections to my ''spanglish'', it could be of much interest for amy Spectrum users, with interest of knowledge about Sinclair hardware.

If you have any news or suggestions , which could be of helpful for me, please, answer to the address below:
Yours sincerely
F. Javier Escribano

Ed's Replay: I can't be of much help, but I wonder if this has anything to do with the reason that the microdrives work perfectly with some Spectrums yet cause problems on others!

TS 2068 to SPECTRUM!

Recently, I converted by Timex TS 2068 computer into a 48K Spectrum. This information may be very useful to your American readers because Spectrum machine code programs will not run very well on the TS 2068, although BASIC programs will. My method s listed below.

I ordered the Spectrum ROM chip form P.V. Tubes, 104 Abbey St., Accrington BB5 1EE, Lancs., England. I enclosed a money order in pounds sterling (available at Barclay's Bank or Deak-Perrar here) for £8.85. This was £5.95 for the chip, £0.90 for VAT, and £2.00 for airmail postage. I also requested that the package be labelled 'DO NOT X-RAY — contains delicate electronic component'. This took two and a half weeks to arrive after I sent the order.

When I received the chip, I opened up the computer. It is important to note which screws came from which holes. I removed the main ROM chip and left the cartridge ROM chip in. The main ROM is in a IC socket next to a small, upside-down circuit board. The cartridge ROM is in another socket near the back of the machine and the printer/expansion port. REMEMBER — The main ROM is near the center of the machine and next to the small upside-down circuit board!!! I strongly recommend using an IC extraction/insertion tool be used to take the main ROM out and put in the Spectrum ROM. It is very important to note on the TS2068 ROM chip that there is a half-circle notch at the top of the chip. There will also be a notch at the top of the Spectrum ROM. The Spectrum ROM must be put in so that the notch faces the same way as the one on the chip taken out, i.e. to the back of the machine. It is also very important to insert the new Spectrum ROM chip so that all the pins are in the correct holes in the socket — i.e. none bent over the edge of the socket and no pins hanging over the top or bottom end of the chip socket. If you feel this is too difficult to do, I recommend asking someone proficient in digital electronics to do it for you. In my own case, I did it myself, but I am also a third year university student of electrical engineering. THIS OPERATION SHOULD BE DONE WITH THE POWER OFF!!!

You now have the equivalent of a 48k Spectrum. You can test this by turning the power on. You should get the Sinclair copyright notice. The new chip will not affect the colours on the TV because the American video circuitry is still there.

It was noted that although the ROM was changed and the systems variables changed (programs don't start at address 23756 but at address 23755) the Input/Output (I/O) hardware is still mapped as before. The following In/Out locations are probably not used on the Spectrum but are still present in the TS 2068: 245,246 — sound synthesizer chip; 255 — advanced video modes.

I believe that there are two locations used for the modem and one for the joysticks, but I haven't been able to find them.

I can basically summarize by saying that it seems to work very well as a Spectrum emulator. Several machine code programs for the Spectrum that use ROM calls work fine. I would be a little cautious about buying English hardware for the machine becuse I am not sure it would work and also the Unite Kingdom power mains are 240 volt 50 cycle system and the US uses a 120 volt 60 cycle system, which would require an expensive converter to power any external hardware.
Sincerely
Paul McGinnis
USA

T/S Users Group

Dear *ZX Computing,*
I am writing to request help on behalf of the thousands of Timex/Sinclair 2068 owners here in the States. We have heard that 2068's are now available in England. Would any software companies which have compatible software for these excellent computers please get in touch with us? Also, any book publishers or booksellers who handle Sinclair computer books can contact us.

Lastly, our local group purchases copies of *ZX Computing* through our local dealer and enjoy it a great deal. Are there other magazines there which we might subscribe to that also cover the Sinclair machines. If so, please send particulars.

Please help us in our hour of need.
Yours,
Richard Cravy

Lost for Words

Dear Sirs,
I have just received as a gift ''speech Synthesiser'' of Computer Add Ons, from my family residing at Leicester.

The above mentioned Model was purchased four months ago. To my astonishment and regret no instructions for use were included, and the company went bankrupt. Therefore, I cannot use the above mentioned and enjoy it's capacities.

I should therefore, be very much obliged to you if you could help me have the suitable Instructions for use.

I do have in my possession a SPECTRUM COMPUTER.
Thanking you in advance, I wish to remain
Yours sincerely
Amit Barak

BASIC on GOTOs

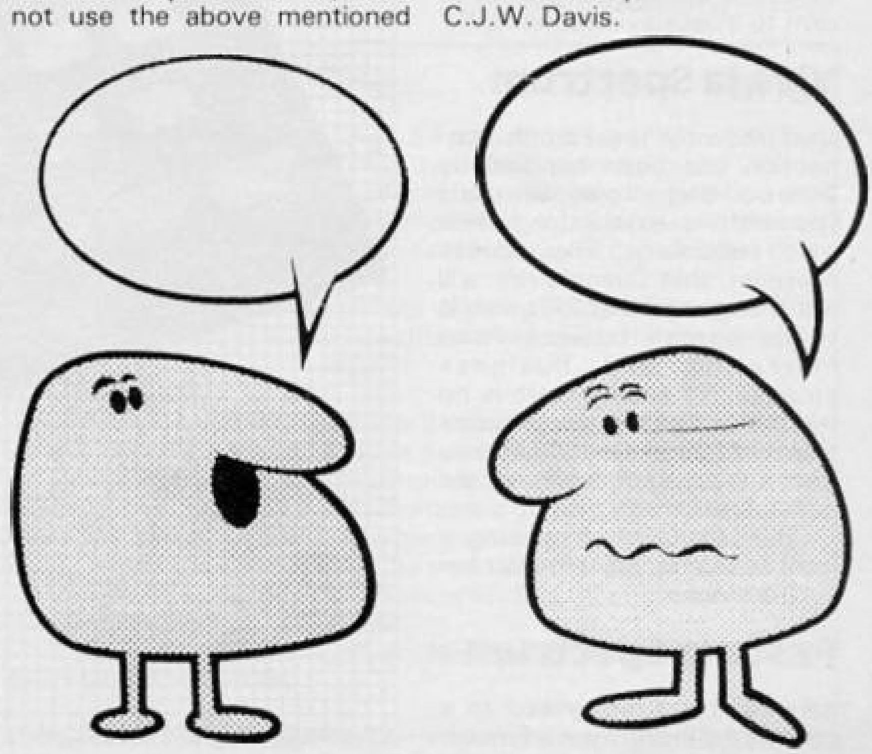
Dear Ray,
Having read the Aug/Sept issue I was intrigued with the toolbox item for the ZX81 of ''on X Goto/Gosub'' by Mark Wenham, giving a machine code routine to do a task missing in the ZX81 commands. However there is a single line of basic that will do this job.

```
1000 GOTO VAL ",0200,
0300, 0400, 0500, etc"
(A * 5-3 TO a 5)
```

This works for Gosub's also, and can be made more compact by omitting the ',' between numbers although harder to read.

```
1000     GOTO     VAL
"0200030004000500
etc"(A * 4-3 TO A * 4)
```

'Time wise' this is slower by .075 of a second (1885 frames per 500 gosubs on my time trial) per gosub/goto and as the most useful application is in response to a user request (and they are notoriously slow) the faster machine code routine does not seem worth the extra 140 byte it uses.
Yours faithfully
C.J.W. Davis.



SHOPTALK

Some of the correspondence and rumours from the industry.

Upward and onward

Since the last issue we have received a constant stream of information from Sinclair Research about their ever increasing overseas outlets.

Alan Miles was appointed as their Middle East manager to develop the already existing market there, while a new office was opened in Singapore for the South East Asia market. John Durnford, ex Commodore employee, has been sent to bring enlightenment to that continent.

Meanwhile, nearer home, H Christensen & Son have been appointed as an exclusive distributor to sell Sinclair products to the Danish market. Interestingly this company are following Sinclair's example and offering a promotional package of software, however it consists of eight programs compared to SR's six.

Even such far flung outposts as Yugoslavia are receiving the tender ministrations of the Sinclair machine. Iskra, one of Yugoslavia's largest electronics firms, are to assemble the Sinclair computers in Ljubljana. A massive TV campaign has been launched there and two thousand Spectrums have been sent to Yugoslavian schools.

Viva la Spectrum

Until recently the French connection has been handled by Direco but now Sinclair Research is establishing their own subsidiary. They stress however, that Direco's role will not change and that SR's area is to be overall business and marketing and business strategy. To ensure there is no friction or clashes the principle shareholders of Direco become joint managing directors of the new subsidiary.

The feasibility of opening a local assembly plant is also being considered.

Pasta la Spectrum

Italy has been recognised as a country suffering from a Sinclair deficiency and Sir Clive has come rushing to the rescue. A new company formed as a joint venture with GBC Italiana will act as a subsidiary office for SR.

Actually my first statement should be taken with a pinch of salt, as GBC have been involved in distributing Sinclair products in Italy for the last ten years!

East Europe acquired the talents of Jan Tyszka as Sinclair Research's business development manager. Jan has some experience of this market and tells us that there is already a considerable amount of locally produced software for the ZX81 and Spectrum.

Meanwhile, back at the ranch. . .
Chris Fox has joined Sinclair Research from Pye and is repsonsible fo the company's full range of flat-screen TV, computer, peripheral and software products.

He also has the strenuous task of liaising with Sinclair's distributors Prism Microproducts and TBD (computers) Ltd.

Well they've got to do somthing with those pre-tax profits of £14.28 million.

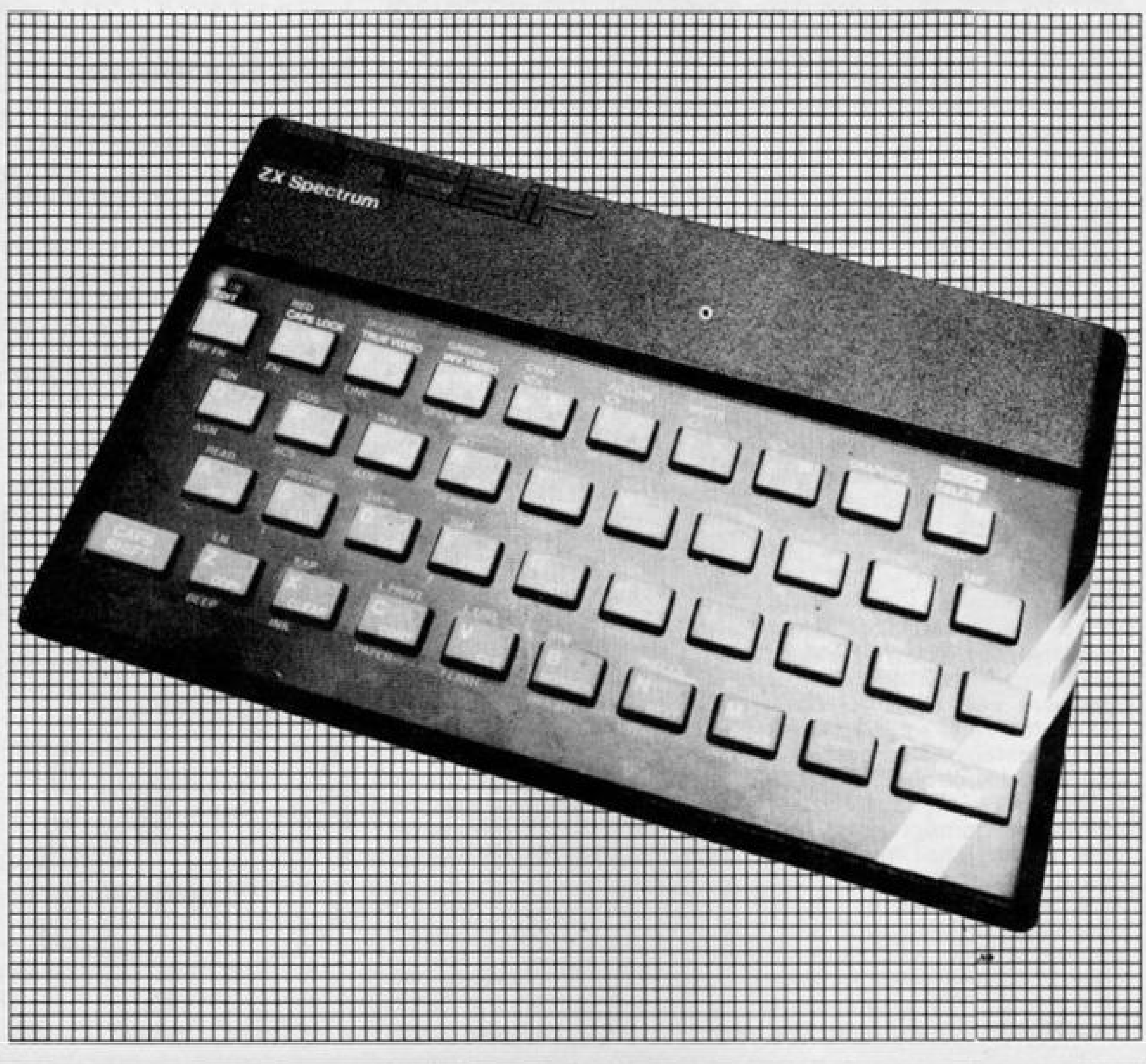
Sinclair to market Imagine Megagames for the QL

Sinclair Research Limited has acquired the rights to **Bandersnatch** – the first of the much publicised "megagames" from recently liquidated Imagine Software Limited – and plans to make it available for the Sinclair QL computer during the first quarter of 1985.

Sinclair also has options to purchase later megagames as they are completed by a new Liverpool based software house, formed by a group of ex-Imagine personnel, including Dave Lawson and Ian Hetherington. The company is working for Sinclair on a fee basis.

According to Alison Maguire, Sinclair's software manager, 'the games have enormous technical and marketing potential. They are at once complex, challenging and very exciting and will be ideal for the type of QL user we envisage.'

Royalty payments from the megagames will go to Imagine's creditors via the official liquidator, Christopher Chambers of Arthur Young, McClelland, Moores & Co. Chamber carried out extensive negotitations before finalising the three part agreement and believes it offers "very significant potential to recover monies for Imagine's unsecured creditors."



NEWS



ZX exchange lives on!

ZX Exchange, soon to celebrate it's second anniversary, continues to flourish, and to provide what is possibly the only remaining international postal forum exclusive to the ZX81 and its overseas counterparts.

The bi-monthly newsletter, "ZX EXCHANGE", in addition to software and programming tips and listings, provides the means to make contact with other users at home and abroad. many useful exchanges have been made between ZX81 users of differnet countries and cultures, with mutually beneficial results. The newsletter also provides information about magazines and other publications of interest to ZX81 users, occasional reviews of soft and hardware, plus news and gossip.

Details of ZX Exchange and a trial issue of the newsletter can be had by sending £1 or (if writing from outside the UK) five international reply coupons, or send a stamped addressed envelope (or one international reply coupon) for further information, to: Nick Godwin, 4 Hurkur Crescent, Eyemouth, Berwickshire, Scotland, TD14 5AP. Telephone: Eyemouth (0390) 50965.

Call back for AGF Protocol 4 Interfaces

AGF are anxious that some early customers who purchased Protocol 4 joystick interfaces may have sub-specification pre-production units.

The problem with these units is that they have an erratic or unrelibale Kempston operation mode and the way to check this would be to type in the following program with the Protocol 4 attached to the Spectrum, its mode switch in the down position and with the 'Kempston' card inserted. Print IN 31, press ENTER and if anything other than 0 is printed, the unit should be returned to AGF for a free replacement.

The address for returns is:

AGF Hardware,
FREEPOST,
Bognor Regis,
West Sussex,
PO22 9BY.

A.G. Fosberry, Proprietor

More from Melbourne House

The company who revolutionised adventure games playing on the computer with the Hobbit were fairly quiet for a while and then in a burst of activity re-emerged to make a further challenge to the market.

In fairly quick succession we were exposed to Mugsy, a graphically superb variation on the strategy/Kingdoms game, Hampstead, an amusing text adventure game and Sherlock which is their top offering.

Sherlock is a complex and different adventure game in which the player has to discover the identity of the criminal and prove it to the inspector. A few strange situations have occured which are suspiciously like bugs (its hard to be really sure in an adventure game), nevertheless it is destined to become a classic — probably by the time this gets into print!

Arcade game players are catered for in the form of Sports Hero. This is a variation on the "Track and Field" arcade hit but is nicely implemented and is well worth trying out at your local software emporium.

Although most Spectrums are now 48K, there is still a significant number of 16K owners who are unhappy about the lack of software being produced for them.

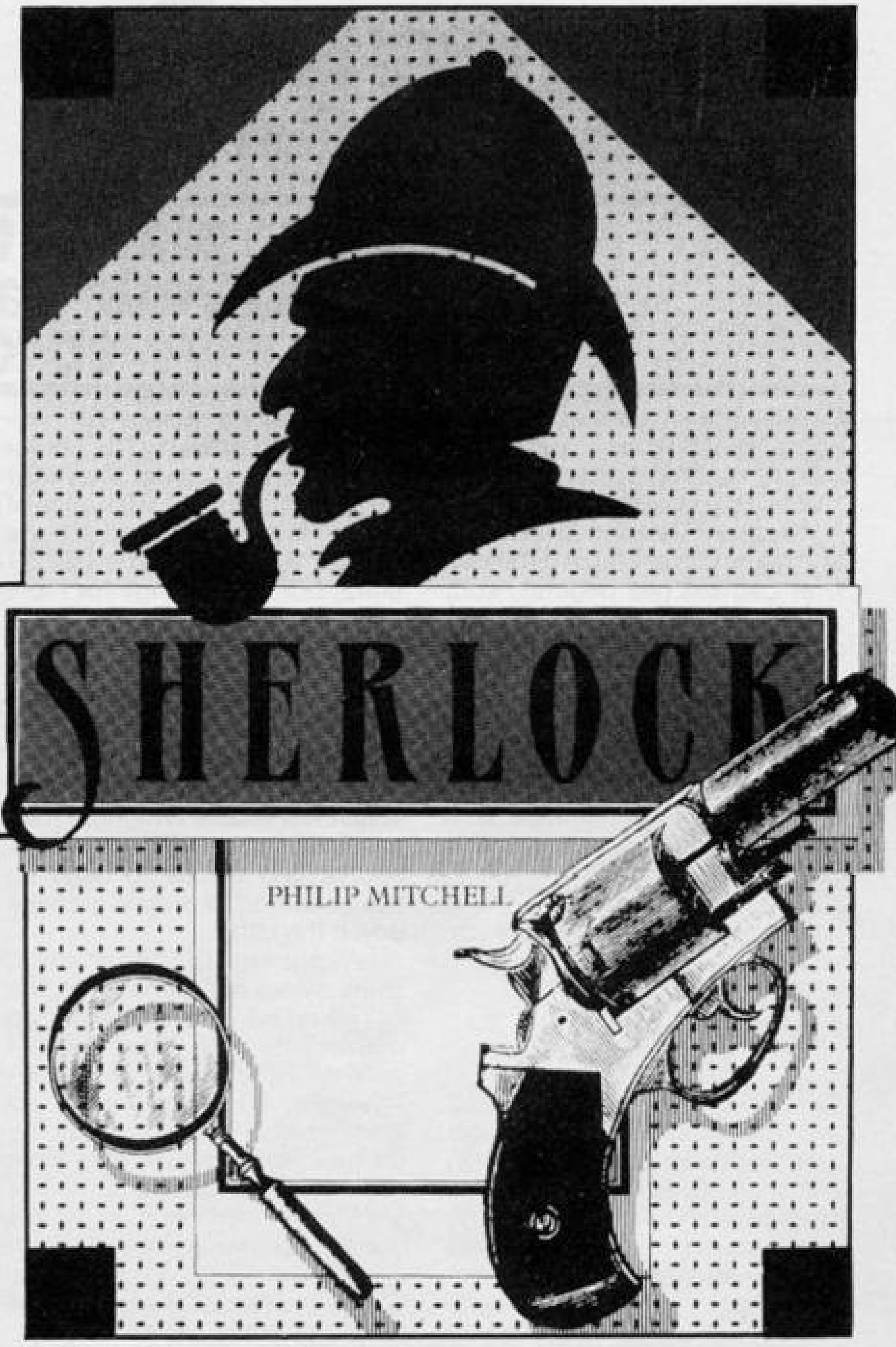
MH's Sir Lancelot shows what is possible in 16K and it is more impressive than some 48K games I've seen. Another one to try out as soon as you can.

Books are another product of this company and they are usually different from the run of the mill computer tome frequently found on the bookshelves.

Learning with Adventure Programs is a must for all teachers, especially those involved with the teaching of English. This book provides detailed outlines of projects undertaken with three adventure games, Snowball, Valhalla and, of course, the Hobbit.

It also gives an evaluation and suggestions for using the Quill in the classroom. Lots of ideas and inspiration for the teacher, but also a lot of preparation time required.

Recommended reading for all programmers who want to develop their ability and do things the way the experts do, is



Spectrum Software Projects.

This book discusses and sets tasks on structured programming techniques, it takes a practical approach and is interesting and lively. Learning to program in this way is a good mental discipline and is valuable for when you move into the more complex world of machine code.

In . . . . Brief

Eidersoft, Freepost, PO Box 54, Ilford, Essex IG1 1BR are marketing a beautiful wall chart type programmers' aid.

Attractive to look at, it contains all the basic information that often takes so long to look up and check using a manual. At only £1.25 I fully recommend them to anyone who uses his/her computer for programming.

This company also specialises in fitting computer keyboards to your specifications to the Spectrum. Ring them for details on 01 478 1291.

Hello and thanks to **The Sensible Shoes**, no this is not an advert for Clarks, but the name of a seven piece band based in Blackpool.

Showing a great deal of enterprise they have produces a record on their own label and it also contains a graphic display and prints the lyrics on your Spectrum while you listen to the music.

Sorry folks, but I must tell you that I liked it! The problem is that nothing I like in music world does well. Still there's always a first time and I wish you all the best.

Copies may be obtained from **Nigel Bernstein**, 232 Hornby Rd. Blackpool, Lancashire.

Beach attack

Mark Jacobs places the country's coastal defences in your hands while he retires to Kent.

Wave after wave of enemy ships pour troops onto the beach, Captain Editor is ready, "We surrender" he cries. Luckily Sergeant Reader is at hand and takes command of defences (the Captain has degate), and, pausing only to type in this program and cover the Captain's yellow streak to prevent the enemy from spotting our position, he orders the first retaliation.

Instructions were dispatched in the program, but here is a terse message containing further details from HQ.
Good luck Sergeant, and why is that man in the Captains uniform waving a white flag?

Guidelines

First type in lines 7000 to 7250 and run it to ensure the user graphics have loaded correctly, and then type in the rest of the program (when doing the cliffs, print a square — "8" in graphics mode — and then the corresponding letter).

When playing the game you can exchange points for ammunition by pressing "5" (your score must be greater than 500 for which you get 20 units of ammunition, and if your score is greater than 1000 you get 30 units, but in exchange for more points).

To make the game harder, when you reach wave 3 your gun will not fire when "8" is pressed, and similarly when wave 5 is reach "9" will not fire the gun. Hence if you miss any ship it will enable two more men to climb up the cliff, unless of course you decide to blow up the beach or to drop a grenade down the cliff.

Whenever "3" is pressed, three mines will be laid on the beach which can only blow up men as they jump out of the boat onto the beach. There is approximately a 40% chance of any man getting blown up as he crosses a mine. At any time there can only be a maximum of 3 mines on the beach, and it is a waste of ammunition by continually pressing "3" since this only achieves a change in the positions of the mines.

Variables

r=	RND
w=	wave number
q=	ammo
n=	speed (counter)
c=	detects presence of men on beach
l=	gun range
f=	wave (counter)
x=	ship (19,x) position
y=	shell position
z=	mine position
v=	man position
k & m=	various loops
z$ =	blow up beach
u=	position of men up cliff
p=	number of guns available
j=	cliff bomb's vertical position

User Graphics

ABC=	ship
F=	man
G=	man explosion
KLM=	ship explosion
N=	cliff bomb
P,Q=	sea graphics
T=	shell splash

CLIFF= DEO RS SU HIH ERSJ in that order (ignore the gaps)

```
   1 REM ***********************
         *Underlined characters*
         *are entered in       *
         *GRAPHICS mode.       *
         ***********************
  20 BORDER 3: PAPER 5: INK 7: CLS
  30 GO SUB 7000
  35 GO SUB 3000
  40 CLS
  60 POKE 23561,255
  65 GO SUB 9000
  70 LET w=1: LET q=75: LET r=RND: LET n=40: LET s=0: LET c=0: LET f=20: LET z=0: LET v=12: LET z1=0: LET z2=0: LET z3=0: LET u=20: LET p=5
  95 LET x=28
 100 PRINT AT 19,x; INK 0;"ABC ": LET r=RND
 110 IF INKEY$="0" AND p>=3 THEN LET l=15: GO TO 300
 120 IF INKEY$="9" AND p>=4 THEN LET l=12: GO TO 300
 130 IF INKEY$="8" AND p>=5 THEN LET l=10: GO TO 300
 140 IF INKEY$="3" THEN  GO SUB 500
 150 IF INKEY$="2" THEN  GO SUB 600
 160 IF INKEY$="1" THEN  GO SUB 900
 165 IF INKEY$="5" AND s>500 THEN  GO SUB 980
 170 FOR i=0 TO n: NEXT i
 172 IF q<1 THEN  GO TO 1500
 178 PRINT AT 1,1; INK 0;"Score"
 180 PRINT AT 1,7; INK 7;INT s;" "
 183 PRINT AT 1,12; INK 0;"Wave"
 190 PRINT AT 1,17; INK 7;w
 193 PRINT AT 1,20; INK 0;"Ammo"
 200 PRINT AT 1,26; INK 7;q;" "
 210 IF c>=1 THEN  GO SUB 700
 225 LET x=x-1
 230 IF x=13 THEN  GO TO 1000
 250 GO TO 100
 300 BEEP .1,-10
```

```
 310 LET y=1*2-INT (RND*7)
 330 IF y=x OR y=x+1 OR y=x+2 TH
EN  PRINT AT 19,x; INK 2;"KLM":
BEEP .1,0: GO TO 360
 340 PRINT AT 19,y; INK 7;"I": B
EEP .07,-2: BEEP .13,4
 345 PRINT AT 19,y;" "
 350 LET q=q-3: LET n=n-2
 351 IF x>20 THEN  LET x=x-1
 355 GO TO 390
 360 PRINT AT 19,x;"   "
 610 PRINT AT 20,2; INK 0;z$
 620 BEEP .6,-8
 630 LET q=q-12
 635 IF c>=1 THEN  LET s=s+25: L
ET c=0
 640 PRINT AT 20,2;"           "
 650 RETURN
 710 PRINT AT 20,k-1; INK 7;"E
"
 718 IF k<=3 THEN  GO SUB 800
 720 LET k=k-1
```



```
 370 LET s=s+50+(RND*10): LET n=
n-2: LET f=f-1: LET q=q-2: GO TO
 1063
 390 IF f<10 THEN  BEEP .05,20:
LET w=w+1: LET f=12+INT (RND*8)
 400 IF w=2 THEN  LET n=35
 410 IF w=3 THEN  LET n=30: LET
p=4
 420 IF w=4 THEN  LET n=25
 430 IF w=5 THEN  LET n=20: LET
p=3
 440 IF w>5 THEN  LET n=1
 450 IF r>=.5 AND r<=.6 THEN  LE
T q=q+20: BEEP .2,0: BEEP .1,5
 455 IF ATTR (19,x)=14 THEN  PRI
NT AT 19,x;"   ": GO TO 95
 460 GO TO 100
 510 LET z1=INT (8+(5*r))
 520 LET z2=INT (6+(7*r))
 530 LET z3=INT (3+(10*r))
 540 LET q=q-4: BEEP .01,10
 550 RETURN
 600 LET z$="GGGGGGGGGG"
 745 LET c=c+1
 770 RETURN
 820 PRINT AT u,2; INK 2;"E"
 825 LET u=u-1
 830 PRINT AT 20,2; INK 2;"E"
 835 IF u<9 THEN  GO TO 1500
 840 LET k=k+9
 845 IF c>=14 THEN  LET c=-3
 850 RETURN
 900 BEEP .1,-4
 910 FOR j=10 TO 20
 920 PRINT AT j-1,2; INK 3;" N"
 930 NEXT j
 935 LET q=q-25: LET u=20
 937 PRINT AT j-1,2;" "
 940 RETURN
 980 BEEP .25,5
 982 IF s>500 AND s<1000 THEN  L
ET s=s-300: LET q=q+20
 985 IF s>1000 THEN  LET s=s-500
: LET q=q+30
 990 RETURN
1000 LET c=1
```

```
1010 FOR k=11 TO 8 STEP -1
1020 PRINT AT 20,k; INK 0;"E": B
EEP .01,0
1030 IF k=z1 AND r>.5 THEN  GO S
UB 1070: LET z1=0
1040 IF k=z2 AND r<.6 THEN  GO S
UB 1070: LET z2=0
1050 IF k=z3 AND r<.8 AND r>.3 T
HEN  GO SUB 1070: LET z3=0
1060 NEXT k
1063 PRINT AT 19,x;"     "
1065 GO TO 95
1070 PRINT AT 20,k; INK 0;"G"
1080 BEEP .05,-5: LET s=s+20
1090 PRINT AT 20,k;" "
1100 RETURN
1500 FOR n=6 TO 0 STEP -1: BORDE
R n: BEEP .15,n: NEXT n: BORDER
3
1510 PRINT AT 1,26; INK 7;"0  "
1512 BEEP .4,0: BEEP .8,5: BEEP
.4,0: BEEP .9,5: BEEP .3,7: BEEP
 1,10
1520 PRINT AT 10,10; INVERSE 1;
FLASH 1; INK 0;"GAME OVER"
1530 PRINT AT 15,16; INK 2;"Anot
her go?(y/n)"
1540 IF INKEY$="y" THEN  RUN 31
1545 IF INKEY$="n" THEN  STOP
1550 IF INKEY$<>"y" THEN  GO TO
1540
3000 FOR i=0 TO 5 STEP .1
3005 PRINT AT 7,9; INK INT i;"BE
ACH  ATTACK"
3010 PRINT AT 10,6; INK 1;"By MA
RK JACOB 1983"
3020 PRINT AT 17,1; INK 2;"Do yo
u want Instructions (y/n)?"
3030 IF INKEY$="y" THEN  GO TO 4
000
3033 IF INKEY$="n" THEN  RETURN
3035 NEXT i
3041 IF INKEY$="" THEN  GO TO 30
00
4000 PRINT AT 2,1; INK 0;"The ob
ject of Beach Attack is  to defe
nd the beach using your  gun on
the cliff.You can fire   at the
ships at different rangesusing t
he 8,9,and 0 keys,which  fire th
e shells with increasing strenth
."
4010 PRINT AT 9,0; INK 0;"If any
 ship comes ashore then   a numb
er of men will commence   their
attack.Random mines can   be lai
d on the beach by pressing3,or t
he whole beach obliteratedby pre
ssing 2."
4020 PRINT AT 15,0; INK 0;"The m
en then start to ascend    the c
liff-their aim is to reach the g
un.Bombs can be dropped    onto
the men by pressing 1"
4030 PRINT AT 20,5; INK 2;"PRESS
 'y' TO CONTINUE"
4040 IF INKEY$="y" THEN  CLS : G
O TO 4050
4045 GO TO 4040
4050 PRINT AT 2,1; INK 0;"Your a
mmo remaining is shown at the to
p of the screen.You lose  ammo a
t the rate given below:"
4060 PRINT AT 6,3; INK 0;"Fire (
hit)..........2"
4062 PRINT AT 8,3; INK 0;"Fire(m
iss)..........3"
4064 PRINT AT 10,3; INK 0;"Lay m
ines...........4"
4066 PRINT AT 12,3; INK 0;"Blow
up beach.......12"
4068 PRINT AT 14,3; INK 0;"Drop
bomb down cliff....25"
4070 PRINT AT 17,1; INK 0;"About
 10% of the ships are      supply
 ships(identical to the   rest),
and when these are hit theammo g
oes up."
4080 PRINT AT 21,15; INK 2;"PRES
S 'y'"
4081 IF INKEY$="y" THEN  CLS : G
O TO 5000
4085 GO TO 4081
5000 PRINT AT 4,3; INK 0;"The ga
me ends when either     the men
reach the gun or you       run ou
t of ammo."
5010 PRINT AT 13,11; INK 1;"GOOD
 LUCK!"
5020 PRINT AT 17,2; INK 2;"PRESS
 'y' TO START THE ATTACK"
5030 IF INKEY$="y" THEN  GO TO 4
0
5040 IF INKEY$<>"y" THEN  GO TO
5030
7000 FOR i=65 TO 85
7005 PRINT AT 9,5; INK 0;"PLEASE
 WAIT A MOMENT"
7010 FOR n=0 TO 7
7020 READ x: POKE USR CHR$ (i)+n
,x
7030 NEXT n
7040 NEXT i
7050 DATA 0,0,0,1,255,127,63,31
7060 DATA 4,142,127,213,255,255,
255,255
7070 DATA 0,0,192,240,255,255,25
5,255
```

```
7080 DATA 255,254,252,252,252,24
8,252,252
7090 DATA 252,254,254,254,254,25
2,252,248
7100 DATA 0,0,24,24,8,24,8,20
7110 DATA 136,34,0,89,24,130,40,
130
7120 DATA 248,248,240,240,240,24
0,248,252
7130 DATA 254,254,254,252,252,25
2,252,248
7140 DATA 248,248,252,252,252,25
4,254,254
7150 DATA 145,37,148,43,76,37,18
,37
7160 DATA 145,36,149,58,204,181,
82,101
7170 DATA 145,36,149,93,216,36,8
2,200
7180 DATA 16,8,16,8,16,8,16,0
7190 DATA 248,241,243,230,236,24
8,240,248
7200 DATA 0,0,192,227,255,255,25
5,255
7210 DATA 24,62,255,255,255,255,
255,255
7220 DATA 248,252,252,252,252,24
8,248,240
7230 DATA 240,240,240,248,248,24
8,248,248
7240 DATA 0,170,0,170,128,84,0,2
4
7250 DATA 252,254,254,254,252,25
2,248,248
7260 CLS
7270 RETURN
9000 PRINT AT 7,0; INK 4;"■D"
9010 PRINT  INK 4;"■E"
9020 PRINT  INK 4;"■O"
9030 PRINT  INK 4;"■R"
9040 PRINT  INK 4;"■S"
9042 PRINT  INK 4;"■S"
9044 PRINT  INK 4;"■U"
9046 PRINT  INK 4;"■H"
9048 PRINT  INK 4;"■I"
9050 PRINT  INK 4;"■H"
9052 PRINT  INK 4;"■E"
9054 PRINT  INK 4;"■R"
9056 PRINT  INK 4;"■S"
9058 PRINT  INK 4;"■J"
9060 PRINT AT 20,14; INK 1;"PQPQ
PQPQPQPQPQPQPQ"
9065 PRINT AT 21,0; INK 6;"█████
████████████████████████████"
9070 RETURN
```

Light Screen Designer

Part 5

Toni Baker continues our machine code mega-program

Before I start Part Five, I'd like to give a brief summary of all the errors which have cropped up in the listing so far. In Part One, the drawing labelled Figure One was incorrect, however this was corrected in Part Two. The other errors are all either errors in the hex listing in the first coloum, or errors in the assembler mnemonics in the remaining colums.

All errors in Part One were corrected in Part Two, except that in the section GET_CHR_2, the instruction INC A should be in the third column, not the second. Part Two however contained errors which were not correct in Part Three. The hex instruction at DD38 should be FEB0, not FEC0. The hex instruction at DD89 should be 1E36, not 1E26. In the section DR_CURSOR, "LD E,DOWN_PIX low" should read "LD E,UP_PIX low", and "LD E,UP_PIX low" should read "LD E,DOWN_PIX low". In Part Three the chunk of code between DE3D and DE56 is all obsolete, and is replaced by a new block of code given in Part Four — which contains an error (which I'll deal with in a minute). Also the hex code at DE67 should be 3807, not 3805, and the hex code at DE94 should be CDB6DD, not CBD6DD. In the section DR_CURSORS the label DSC_CONT should be one line down, and further on, "RET" should be in the third column, not the second. In the section, START, the label "LSD_LOOP" is in the wrong place. It is currently in the third column — it should in fact be in the second column, and one line up. The instruction LD E,(HL) should be in the third column (where "LSD_LOOP" is now) and not in the fourth column. In the section MAIN_LOOP, "LD DE,(CURSOR+2)" should read "LD BC,(CURSOR+2)"; The lables "CSR_DOWN" and "CSR_UP" should be transposed, on the same lines "CALL UP_PIX" and "CALL DOWN_PIX" should also be transposed, and also the accompanying notes.

Yet More . . .

Now onto Part Four, where we find an error in the ammendments to Part Three. In the section ML_TEST there is a missing line, which should be positioned directly ***above*** the label ML_ACTION. The missing line should read "CBAE — RES 5,(HL)". Also in Part Four; at address DEC2 the word "UNDO" appears in the first column amongst the hex code — this is in fact a label, and hence should appear in the second column. At address DED8: "2106DB" should read "210CDB", and at address DEFE: "CREADE" should read "CDEADE". In the section MOVE, "LD (ORIGIN+2), BC" should be in the third column, not the second, and the notes "Change origin. . ." etc on the same line should be in the fourth column, not the third.

Phew . . .

Anyway, this issue's article covers thirteen more keys, and thus enables you to do a whole lot more. From the top row of the keyboard we have the functions STORE and RECALL, and also CLS. (Note that CLS is on key "9", not on key "V" as in BASIC). From the second row the function CLEAR — this is not the same as the BASIC CLEAR, but I'll explain what it does in this program in a moment. From the third row we have the function USR, and on the fourth row — every single key will now be operative.

Store And Recall

STORE is activated by key "3", and RECALL by key "4". The program uses ten "memories" — numbered from zero to nine. Each memory can store the position of the cursor. In other words, if you type STORE 5, then the cursor position will be stored in memory 5. If you, at some later stage type RECALL 5, the cursor will move to the "remembered" position. In addition to merely storing cursor positions, you can STORE and RECALL the whole screen. Typing STORE 'enter' will store the entire screen picture (and the positions of all three cursors). Typing RECALL 'enter' will restore the screen to the stored picture, and restore the cursors too. All of the memories are initially designated "unused". Attempting to RECALL an unused memory produces no effect. Once a memory has been used, however, it is not possible to declare it "unused" any more. The SCREEN memory, however, CAN at any stage be declared unused, simply by pressing the button "CLEAR" (key "I"). Note that all the time there is an image stored in the screen memory the function UNDO will not work, so it is advisable to CLEAR the screen memory as soon as you have finished using it.

CLS (key "9") will clear the screen in the current colours. You can change these current colours at will (see below) and hence you can clear the screen in any colours.

The function **USR** is more or less self explanatory. Its purpose is to call a machine code subroutine at any address. You must type "USR" followed by four hex digits (forming a machine code address), followed by *enter*. A subroutine at that address will then be called. If you type in an address wrongly you can of course press



the ESCAPE button (SPACE) in order to cancel your decision. As examples, USR 0D6B will clear the screen, and USR 1CEE will generate a STOP statement report. the purpose of the function is of course to make Light Screen Designer infinitely flexible. You can write your own machine code programs to perform various graphic functions, and call them from within LSD. (Note that in keeping with convention, such a program should always use the SPACE key as an ESCAPE function — that is, it must cancel the current operation and return to Light Screen Designer). It is of course possible to use your own BASIC

as well — simply press ESCAPE to return to BASIC, carry out whatever tasks you require, and then use USR 56789 to re-instate Light Screen Designer.

COPY makes a copy of the screen image onto the ZX printer. It is the same as the COPY command in BASIC.

PAPER and **INK** change the current paper and ink colours as you would expect. Note that PAPER 8 (transparent) and PAPER 9 (contrast) are both allowed. **FLASH** and **BRIGHT** may be switched on (1) and off (0) at will. Note that FLASH 8 (transparent) is also allowed. **BORDER** changes the colour of (a) the border, and (b) the lower two lines of the screen. **OVER** and **INVERSE** may be switched on and off at will — eg to "un-PLOT" a point just type INVERSE 1, PLOT and INVERSE 0. All of these colour controlling functions are on the same keys as in BASIC, except for BRIGHT, which is on the "CAPS SHIFT" key.

And there we have it. With the exception of CLEAR, all of this issue's functions require the user to INPUT something after selecting the function — either to select a memory number, a colour, a machine code address, or just to say "Y" or "N" (yes or no). All of this is made possible by the MESSAGE subroutine given in Part One. Go back to it if you like, to see how it works. Another point of interest about this Part is that all of the colour functions (except BORDER) are all handled by the same routine — ie there is but one subroutine which copes with PAPER, INK, FLASH, BRIGHT, OVER and INVERSE. This lies at address DF5D.

In order to associate the various subroutines to the various keys, it is necessary, once again, to update the address table. The address table update routine is given as figure 1.

The new bits of program cover addresses DF3E to E01B (see figure 2). We start off with a subroutine called LOCATE, which is used in STORE and RECALL, and then move onto a table, called COL_TABLE, which is used by the colour functions. In Part Six of this series I'll get down to some geometry — all this drawing rectangles and stuff.

See you then — bye bye.

```
DB42  Ø4 EØ          DEFW EØØ4, BORDER
DB44  52 ØØ          DEFW ØØ52, CIRCLE CENTRE  (not yet defined).
DB46  52 ØØ          DEFW ØØ52, QUARTER ELLIPSE (not yet defined).
DB48  ØØ ØØ          DEFW ØØØØ, not used.
DB4A  ØØ ØØ          DEFW ØØØØ, not used
DB4C  52 ØØ          DEFW ØØ52, DRAW_ANR  (not yet defined).
DB4E  52 ØØ          DEFW ØØ52, CIRCLE THRU  (not yet defined).
DB5Ø  5D DF          DEFW DF5D, FLASH
DB52  5D DF          DEFW DF5D, OVER
DB54  52 ØØ          DEFW ØØ52, RECTANGLE  (not yet defined).
DB56  52 ØØ          DEFW ØØ52, PARALLELOGRAM (not yet defined).
DB58  ØØ ØØ          DEFW ØØØØ, not used
DB5A  9B DF          DEFW DF9B, RECALL
DB5C  52 ØØ          DEFW ØØ52, DRAW AU  (not yet defined).
DB5E  52 ØØ          DEFW ØØ52, ELLIPSE  (not yet defined).
DB6Ø  5D DF          DEFW DF5D, PAPER
DB62  5D DF          DEFW DF5D, INVERSE
DB64  ØØ 52          DEFW ØØ52, TRIANGLE (not yet defined).
DB66  8Ø DF          DEFW DF8Ø, CLEAR
DB68  ØØ ØØ          DEFW ØØØØ, not used.
DB6A  85 DF          DEFW DF85, STORE
DB6C  52 ØØ          DEFW ØØ52, DRAW AT (not yet defined).
DB6E  38 DF          DEFW DF38, CANCEL MARK
DB7Ø  5D DF          DEFW DF5D, INK
DB72  5D DF          DEFW DF5D, BRIGHT
DB74  C2 DF          DEFW DFC2, USR
DB76  24 DF          DEFW DF24, CURSOR TYPE
DB78  B8 DF          DEFW DFB8, CLS
```

```
DB7A  52 ØØ          DEFW ØØ52, FILL (not yet defined).
DB7C  Ø4 DF          DEFW DFØ4, DRAW LINE.
DB7E  3Ø DF          DEFW DF3Ø, MARK
DB8Ø  ØF EØ          DEFW EØØF, COPY
DB82  A4 DE          DEFW DEA4, ESCAPE
DB84  28 DF          DEFW DF28, HIDE
DB86  52 ØØ          DEFW ØØ52, TEXT (not yet defined).
DB88  BE DE          DEFW DEBE, UNDO
DB8A  52 ØØ          DEFW ØØ52, PAINT (not yet defined).
DB8C  FB DE          DEFW DEFB, PLOT
DB8E  F3 DE          DEFW DEF3, MOVE
```

```
                           ORG DF3E
Ø1Ø4ØØ     LOCATE          LD BC,ØØØ4          BC:= number of bytes per memory.
1114DB                     LD DE,CURSOR        DE: points to cursor address.
87                         ADD A,A
87                         ADD A,A
C658                       ADD A,58
6F                         LD L,A
62                         LD H,D              HL: points to required memory.
C9                         RET


                           ORG DF4B
5ØØD12     COL_TABLE       DEFB 5Ø, ØD, 12     Information concerning FLASH.
52ØF15                     DEFB 52, ØF, 15     Information concerning OVER.
6ØØC11                     DEFB 6Ø, ØC, 11     Information concerning PAPER.
621Ø14                     DEFB 62, 1Ø, 14     Information concerning INVERSE.
7ØØB1Ø                     DEFB 7Ø, ØB, 1Ø     Information concerning INK.
721113                     DEFB 72, 11, 13     Information concerning BRIGHT.
                           ORG DF5D
214BDF     COLOURS         LD HL,COL_TABLE     HL: points to colour information table.
45                         LD B,L              BC:= large number.
EDB1                       CPIR                Locate required entry in table (note that
                                               A contains low part of pointer into
                                               address table).
116FDF                     LD DE,COL_MES
EDAØ                       LDI                 Alter program to print required message.
1E77                       LD E,COL_CTR low
EDAØ                       LDI                 Alter program to          select
                                               appropriate function.
CDCCDC                     CALL DCCC,MESSAGE
ØØ         COL_MES         DEFS Ø1                Print required message.
F5                         PUSH AF                Stack user's response.
3EØ2                       LD A,Ø2
CDØ116                     CALL 16Ø1,CHAN_OPEN    Select stream two (the screen).
3EØØ                       LD A,??                A:= required control character.
           COL_CTR         $-1
D7                         RST 1Ø                 Print control code.
F1                         POP AF                 A:=             user's response.
D63Ø                       SUB 3Ø                 A:= required colour.
D7                         RST 1Ø                 Select required colour.
C3AD1C                     JP 1CAD,PERMS_SET      Make colours    permanent.
```

```
                    ORG DF8Ø
DDCBØ1A6  CLEAR     RES 4,(J_FLAGS)high   Free memory for use by UNDO.
C9                  RET
CDCCDC    STORE     CALL DCCC,MESSAGE
Ø4                  DEFB Ø4               Print message "Store memory".
FEØD                CP "enter"
28Ø7                JR Z,ST_SCR           Jump if "enter" pressed.
CD3EDF              CALL DF3E,LOCATE      Locate required memory.
EB                  EX DE,HL
EDBØ                LDIR                  Store cursor position.
C9                  RET
DDCBØ1E6  ST_SCR    SET 4,(J_FLAGS)high   Signal "memory is being used".
C3C9DE    S/R_SCR   JP DEC9,COPY_SCREEN   Store screen (note Z flag is set).
CDCCDC    RECALL    CALL DCCC,MESSAGE
Ø5                  DEFB Ø5               Print message "Recall memory".
FEØD                CP "enter"
28ØE                JR Z,REC_SCR          Jump if "ønter•    pressed.
CD3EDF              CALL DF3E,LOCATE      Locate required memory.
E5                  PUSH HL               Stack memory address.
Ø9                  ADD HL,BC             Point   to next memory.
2B                  DEC HL                Point to last byte of current memory.
7E                  LD A,(HL)             A:= last byte of memory.
E1                  POP HL                HL: points to required memory.
FEBØ                CP BØ
DØ                  RET NC                Return if memory not in use.
EDBØ                LDIR                  Recall memory.
C9                  RET
DDCBØ166  REC_SCR   BIT 4,(J_FLAGS)high
2ØE1                JR NZ,S/R_SCR         Recall screen if memory in use.
                                          (note Z flag is reset).
C9                  RET
```

```
                         ORG DFB8
CDCCDC    CLEAR_SCREEN   CALL MESSAGE
Ø6                       DEFB Ø6                Print message "Clear screen?".
FE59                     CP "Y"
CA6BØD                   JP Z,ØD6B,CLS          Clear screen if required.
C9                       RET

                         ORG DFC2
CDCCDC    USR_CALL       CALL MESSAGE
Ø9                       DEFB Ø9                Print message "Machine code address?".
21Ø1ØØ                   LD HL,ØØØ1
E5        USR_LOOP       PUSH HL
FDCB3ØDE  USR_KEY        SET 3,(FLAGS2)         Engage CAPS LOCK.
CDDE15                   CALL 15DE,WAIT_KEY1    Wait for key to be pressed.
FE2Ø                     CP 2Ø
282C                     JR Z,USR_ESC           Jump if ESCAPE pressed.
CD882C                   CALL 2C88,ALPHANUM
```

```
3ØFØ                JR NC,USR_KEY         Jump unless letter or digit pressed.
FE47                CP "G"
3ØEC                JR NC,USR_KEY         Jump if "G" to "Z" pressed.
F5                  PUSH AF
D7                  RST 1Ø                Print hex digit.
F1                  POP AF
D63Ø                SUB 3Ø                Codes "Ø" to "9" now correct.
FEØA                CP ØA
38Ø2                JR C,USR_C            Jump if digit "Ø" to "9".
D6Ø7                SUB Ø7                Codes "A" to "F" now correct.
```

```
E1        USR_C       POP HL                HL:= "address so far".
29                    ADD HL,HL
29                    ADD HL,HL
29                    ADD HL,HL
29                    ADD HL,HL             Shift HL one digit left.
F5                    PUSH AF               Stack the carry flag.
B5                    OR L
6F                    LD L,A                Incorporate new digit.
F1                    POP AF                Restore carry flag.
3ØD5                  JR NC,USR_LOOP        Loop back until digits collected.
E5                    PUSH HL               Stack m/code subrt address.
CDDE15    USR_ENTER   CALL 15DE,WAIT_KEY1   Wait for key to be pressed.
FE2Ø                  CP 2Ø
28Ø5                  JR Z,USR_ESC          Jump if ESCAPE pressed.
FEØD                  CP "enter"
2ØF5                  JR NZ,USR_ENTER       Jump back unless "enter" pressed.
E5                    PUSH HL               Place dummy entry on stack.
E1        USR_ESC     POP HL                Delete item from stack (either dummy
                                            entry or machine code address).
1815                  JR UBC_LOWER          Jump to erase message from lower
                                            part of screen.
```

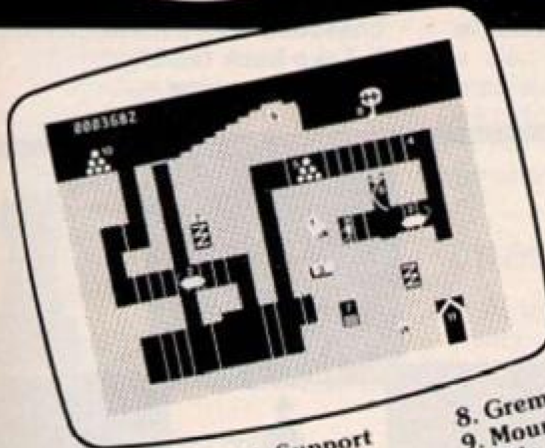
```
                        ORG BØØ4
CDCCDC    BORDER_COL    CALL MESSAGE
ØE                      DEFB ØE               Print message "Border colour?".
D63Ø                    SUB 3Ø                A:= required border colour.
CD9722                  CALL 2297,BORDER_A    Change border colour.
18ØA                    JR UBC_LOWER          Jump to change colour of bottom
                                              two lines of screen.
CDCCDC    COPY_SCREEN   CALL MESSAGE
ØA                      DEFB ØA               Print message "Copy?".
FE59                    CP "Y"
CØ                      RET NZ                Return if reply was "N".
CDACØE                  CALL ØEAC,COPY        Copy screen onto printer.
C36EØD    UBC_LOWER     JP ØD6E,CLS_LOWER     Reset lower part of screen.
```

LOOK!

NOW THERE ARE 3

HI-RES PROGRAMS FOR THE 16K ZX-81

SOFTWARE FARM



FORTY NINER

In 1849 the Great American Gold Rush started. Almost everyone who could sold up everything and dashed to the west coast to look for this precious metal – including you!

You must excavate this precious metal – but can you survive the giant rats and that vicious Gremlin which will come to infest your mine? Can you trick the snakes into leaving their comfortable nests and destroy the rats for you? Can you keep the Gremlin at bay?

Riches await you – but so do the hazards!

ROCKET MAN

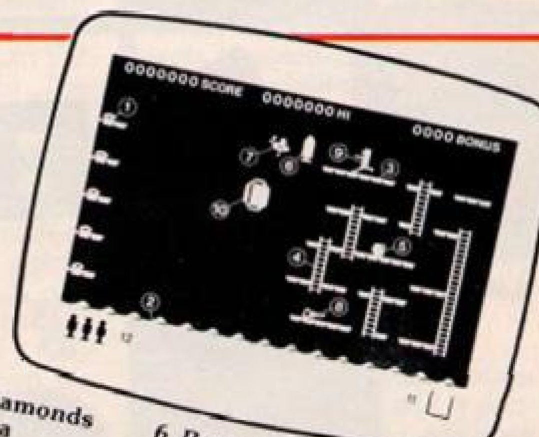
Get rich quick by collecting Diamonds that are simply lying there waiting for you!

Oh . . . I forgot to mention that there are one or two problems!

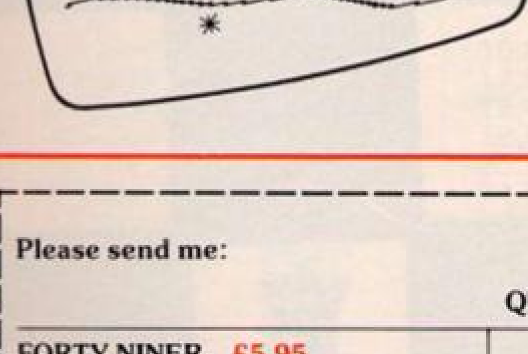
There is an expanse of shark infested water between you and the Diamonds and a strange breed of Bubble that seems hell bent on getting you in it! Somehow you must cross it . . . .

You have a Rocket Pack to help you (a Vulture on higher levels) but you must rush around the platforms and ladders collecting cans of fuel (legs of lamb with the Vulture) and cursing that weird Bubble. Once you have enough fuel then it's Chocks Away!

Oh . . . but don't run out of fuel on the way – otherwise it's . . . SPLASH!



new-new-new-new-new



Z-XTRICATOR

A long time ago, in a galaxy far, far, away a terrible war took place between two hostile races. Any prisoners taken could not expect to live very long in the hands of their captors. Their only hope lay with a group of valiant warriors – the XTRICATORS – whose task it was to rescue fellow beings from the alien planet's surface. You are about to take on the role of such a warrior . . . .

Please send me:

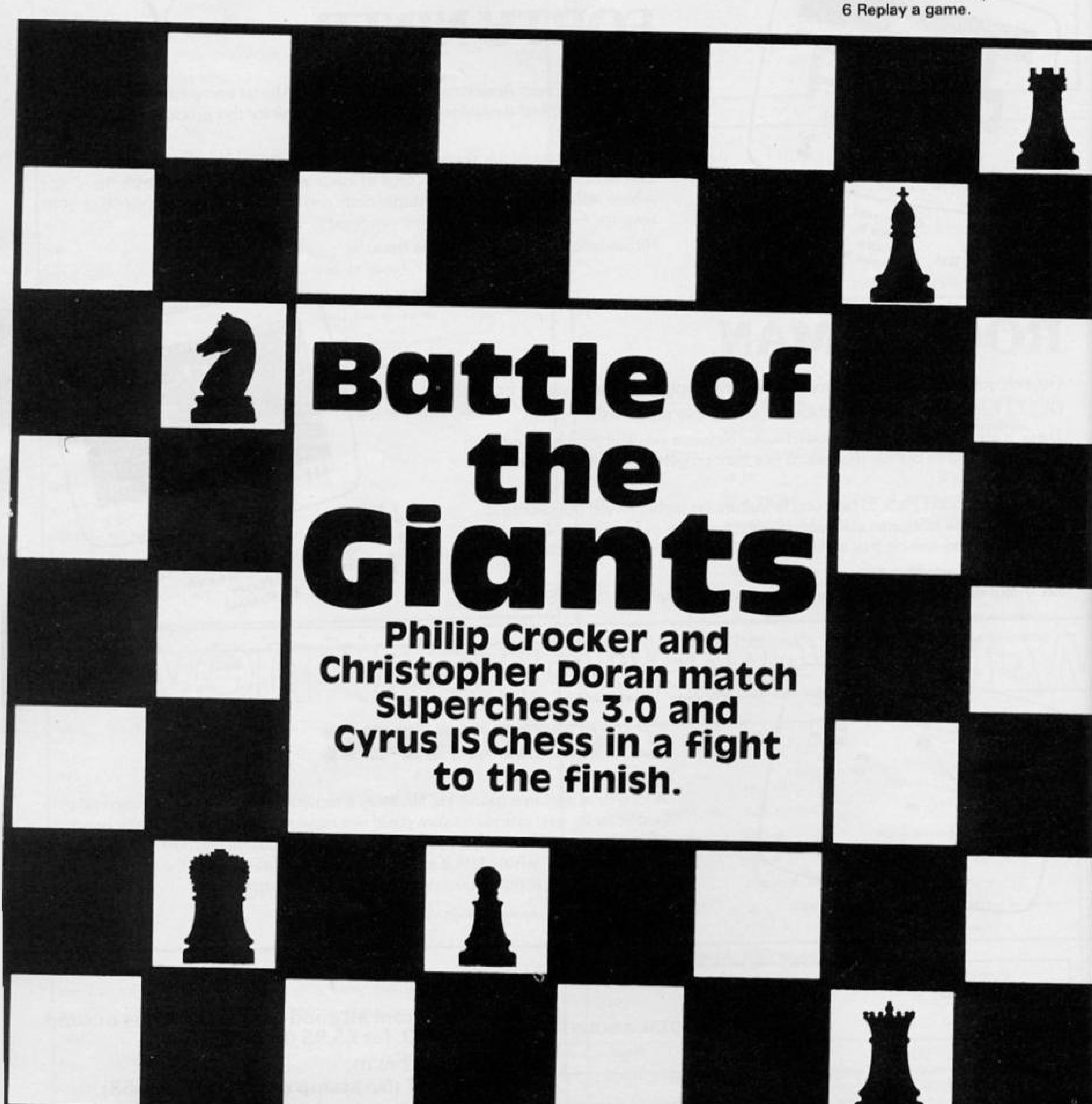
	QTY	TOTAL AMOUNT
FORTY NINER £5.95		
ROCKET MAN £5.95		
Z-XTRICATOR £5.95		
TOTAL		

ZX

Available from all good computer shops or send cheque/P.O. for £5.95 (inc. P&P) to:

Software Farm,
FREEPOST (No stamp required) (BS3658),
BS8 2YY.

Software Farm, 155 Whiteladies Road, Clifton, Bristol BS8 2RF. Telephone (0272) 731411. Telex 444742 AFMADV G



Battle of the Giants

Philip Crocker and Christopher Doran match Superchess 3.0 and Cyrus IS Chess in a fight to the finish.

Since the Spectrum first came out there have been many chess programs written for it. As the Spectrum has become more popular there has been a greater incentive to write the best Spectrum chess program.

The two most obvious contenders for title best Spectrum chess program are Cyrus IS Chess and Superchess 3.0 by CP Software. Cyrus IS Chess is based on a program which won the 1981 European Microcomputer Chess Championship. While Superchess 3.0 comes with a guarantee from CP Software, that it will beat any other Spectrum chess program.

Cyrus IS Chess versus Superchess 3.0

Both chess programs come in a well presented package which contains a cassette and an instruction booklet.

Cyrus contains two programs: one for the 16K and one for the 48K Spectrum. Both Cyrus programs have the following features:

1 Eight playing levels.
2 Cursor key movement of pieces.
3 Auto-demonstration mode.
4 It can be used as a chessboard by two humans.
5 Take-back facility.
6 Replay a game.

7 Set up a position.
8 It solves chess problems (up to mate in three).
9 Turn round board.
10 Change sides during a game.

The additional features for the 48K Spectrum are:

1 Full library of chess openings.
2 Change colour of board and pieces.
3 Print out a record of the game.
4 Save game on tape (for future continuation).
5 Force the computer to make its best move.

Superchess 3.0 is for the 48K Spectrum only. Its features are:

1 Ten playing levels.
2 Recommended move option.
3 Change sides and level of play during game.
4 Set up a position.
5 Self-play option.
6 Technical information (how the program works). During the game Superchess gives itself a score dependent on positional factors and material advantage or loss.
7 Library of opening moves.
8 Change colour scheme.
9 Solve chess problems (up to mate in four).
10 List Moves.

Cyrus definitely has more features than Superchess 3.0. For example Cyrus's take-back facility is a very useful feature which Superchess lacks. There is also a replay feature which enables the beginner and expert alike to examine the previous game, and even to begin the game at a chosen point.

Chess Ability

In order to discover the chess ability of both programs we decided to play the two programs off against each other. We attempted to play the two chess programs off at approximately equal time limits. The results are shown in table 1.

Example Game

This game was between Cyrus (white) and Superchess (black). Cyrus played on level 6 and Superchess played on level 3. The game was typical of the games which were played. Cyrus played a fine opening due to its superior opening knowledge dand the game soon reached a middlegame with a positional advantage to Cyrus . Cyrus exploited this advantage with some fine, logical play and won a pawn. However this led to a multitude of exchanges and an endgame was reached with Cyrus having some winning chances and almost certain drawing chances.

However in the endgame Superchess came into its own. Its driving, delicate play forced zugzwang (the compulsion to make a move resulting in material or positional loss) upon Cyrus. Superchess then proceeded to overwhelm the depleted Cyrus and quickly concluded the game. The moves are listed in table 2.

For those of you who are not familiar with the notation used in table 2, I will brieflyexplain how it works. The piece to be moved is shown by the first letter of its name except for the knight which is represented by N. There is then a dash, or an x if the move captures a piece. The square to which the piece should be moved is shown by the piece which would be on the same file at the beginning of a game and then by the number of squares forward (one to eight). The symbols are KR (kings rook), QR (queen's rook), QN, KN, Q, K, QB and KB. Kingside castling is shown by 0-0 and queenside castling by 0-0-0. Check is shown by + and ! indicates a good move. If for example two rooks could move to the same square then the correct rook is shown in brackets. The moves should be worked out by looking at the board from the point of view of the person who is moving.

White	Black	Result	Moves
Cyrus Level 1	Superchess Level 0	½-½	35
Superchess Level 0	Cyrus Level 1	1-0	64
Cyrus Level 4	Superchess Level 1	1-0	25
Superchess Level 1	Cyrus Level 4	1-0	57
Cyrus Level 5	Superchess Level 2	0-1	38
Superchess Level 2	Cyrus Level 5	½-½	30
Cyrus Level 6	Superchess Level 3	0-1	60
Superchess Level 3	Cyrus Level 6	½-½	65
Cyrus Level 5	Superchess Level 1	½-½	57
Superchess Level 1	Cyrus Level 5	0-1	65
Cyrus Level 7	Superchess Level 6	0-1	110
Superchess Level 6	Cyrus Level 7	½-½	81

Game	1	2	3	4	5	6	7	8	9	10	11	12
Superchess	½	1	0	1	1	½	1	½	½	0	1	½
Cyprus	½	0	1	0	0	½	0	½	½	1	0	½

Table 1. Results from IS Chess v Superchess play off.

Conclusion

Superchess was the clear winner in this contest and this supports the claim that it can beat any other Spectrum chess program. Superchess seems to improve as the game goes on. Its standard opening play and weak (compared to Cyrus) opening library convey a dubious impression. Its middlegame is sound without being spectacular but its endgame is excellent and this is where it won or drew games in which it had a material or positional disadvantage. For example in the final game Cyrus reached the endgame three pawns up and with a better position after a brilliant combination. However by playing outright blunders which lost Cyrus material and decentralised its pieces it enabled Superchess to draw the game with some subtle play.

Cyrus was rather a disappointment in the game just described. Often it would play an excellent opening and middlegame and then lose a "won" endgame. Sometimes on lower levels it would play nonsensical moves in the middlegame or in the endgame. For example it would move a rook backwards and forwards for several moves which enabled Superchess to improve its position unhindered.

Both packages have room for improvement. Superchess has fewer features than Cyrus and it also has a weak opening library compared to Cyrus. Cyrus has weak endgame play and it has a habit of making nonsensical moves.

Cyrus IS Chess is written by Intelligent Software and is priced £9.95. It can be obtained from most computer software retailers. Superchess 3.0 costs £8.95 and can be obtained from the following address:

CP SOFTWARE, Dept YC2
2 Glebe Road,
Uxbridge
Middlesex UB8 2RD

It may also be available from good software shops.



Table 2. The Cyrus versus Superchess example game.

1.P-K4	P-QB4	16.R-N1	0-0	31.K-K3	P-R4	46. K-Q2	P-B8(Q)
2.P-KB4	N-QB3	17.B-QB3	K-R1	32.P-R4	K-N3	47.KxQ	K-B6
3.N-QB3	P-Q3	18.0-0-0	P-QR4	33.P-Q4	K-82	48.K-Q1	KQ6!
4.B-Qn5	B-Q2	19.R-N4	P-R5	34.P-Q5	PxP	49.K-B1	K-85
5.N-B3	N-B3	20.R(Q)N1	R-N8	35.PxP	K-K2	50.K-Q2	KxP
6.P-Q3	P-QR3	21.RxB!	RxR	36.P-N3	P-R6	51.K-Q3	K-B4
7.BxN	BxB	22.BxBP	R(R)N1	37.K-Q3	P-N4!	52.K-B3	P-N5+
8.Q-K2	P-K3	23.R-N5	P-QN3	38.P-N4	P-B5	53.K-Q3	P-N6
9.B-K3	P-Q4	24.K-N1	P-KR3	39.K-K3	K-Q2	54.PxP	P-R7
10.PxP	NxP	25.R-N4!	K-R2	40.P-Q6	KxP	55.K-B2	P-R8(Q)
11.NxN	BxN	26.BxR	RxR	41.K-Q4	Kb3!	56.K-Q3	Q-Q5+
12.B-Q2	Q-KB3	27.P-QB4	P-KB4	42.K-B3	K-Q4	57.K-B2	K-N5
13.Q-K5	BxN	28.RxR+	KxR	43.K-B2	K-75	58.K-N1	Kxp
14.QxQ	PxQ	29.K-B2	K-B3	44.K-Q2	P-B6+	59.K-B1	Q-Q5
15.PxN	B-N2	30.K-Q2	K-B2	45.K-K2	P-B7	60.Q-Q8	mate

THE AGE OF THE R.A.T



CONVENTIONAL JOYSTICKS ARE DEAD!

*The Cheetah **R**emote **A**ction **T**ransmitter is the most sophisticated computer controller available.*

It has these features:

- *Infra Red transmission – so there are no leads trailing across the living room. Just sit back in your chair up to 30 feet from your machine.*
- *Touch control – no moving parts, extremely fast, long life.*
- *No extra software required.*
- *Can be used with all Cheetah RAT/Kempston compatible software.*
- *Fits comfortably in your hand for long play periods.*
- *Comes complete with receiver/interface unit which simply plugs into the rear of your Spectrum.*
- *Compatible with all Sinclair/Cheetah peripherals via the rear edge connector.*

Simply incredible at £29.95 including VAT and p&p.

Dealer enquiries welcome. Export orders at no extra cost.

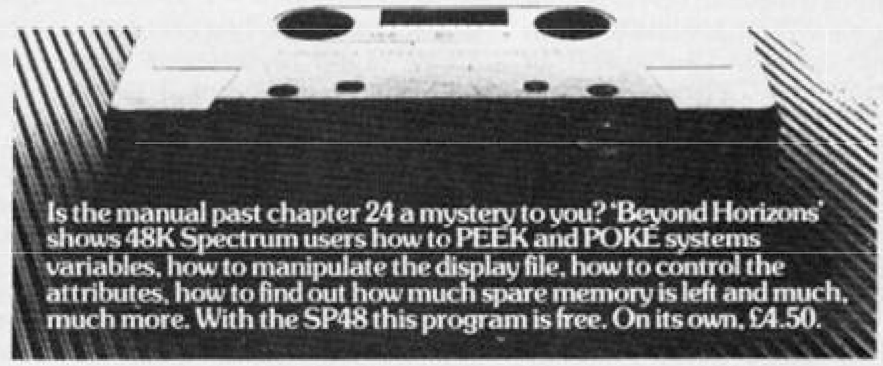
Send cheque/p.o now to:
Cheetah Marketing Ltd.** (Dept. ZX), **24 Ray Street, London EC1R 3DJ. phone 01·833 4909

Cheetah products are also available from branches of
John Menzies Boots WHSMITH Rumbelows
and all good computer shops.

Cheetah Marketing

Ask for our products at your local computer shop

Extend your 16k Spectrum to 48k for just £27·00. And get a Free program worth £4·50 into the bargain.



SP48B, 32k Memory Extension with Program — £27

The SP48 simply plugs into the sockets provided by Sinclair inside your Spectrum and turns your 16k machine into an absolutely standard 48k ready to run 48k programs.

The SP48 fits Issue 2 and Issue 3 Spectrums. Full step-by-step instructions are provided. Fitting is easy and there is no soldering.

The SP48 carries our full warranty and all our customers enjoy the benefits of our renowned product support.

And you get a free copy of Beyond Horizons, and educational program which has already enlightened thousands of 48k Spectrum owners about the inner workings of their computers.

48/80 FORTH

This compiler and editor is fast becoming recognised as the most professional and highest quality implementation of the FORTH language on the Spectrum. FORTH gives you the speed of machine code without the tedium of machine-code programming.

All the Spectrum's excellent graphics commands are retained including DRAW, PLOT and CIRCLE. Real arcade quality sounds can be produced using BEEP.

More advanced FORTH programmers will find that 48/80 FORTH provides a complete implementation of FIG-FORTH, including the sophistications of BUILDS, DOES and CODE.

Each 48/80 FORTH cassette includes a separate FORTH editor which uses part of RAM as if it were disk. The 48K Spectrum allows a 16k RAM 'disk', while leaving nearly 20k of dictionary space. A comprehensive user-manual is also included, which covers both compiler and editor, and has a lucid beginners introduction.

Extensions to 48/80 FORTH will shortly be available to existing customers at a nominal charge. These will include floating point arithmetic, and microdrive handling.

TRADE ENQUIRIES WELCOME — Are you on our trade/export mailing list?

All products on this order form are in stock. Same day dispatch for phoned Access and Visa orders.

To East London Robotics Ltd; Please send by return (tick items required)

☐ SP48B (for Spectrum Issues 2 or 3) £27 ______

☐ BEYOND HORIZONS (48k) £4.50 ______

☐ 48/80 FORTH (48k or 80k) £14.95 ______

Postage 0.65p. I enclose a cheque/PO for ______ My Access/Visa No. is

☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐☐

Name ______

Address ______

ZX1

East London Robotics Limited, St. Nicholas House, The Mount Guildford, Surrey GU2 5HN.

Orders and special enquiries Guildford (0483) 505605

Telex 859428 (Trade only)

EAST LONDON ROBOTICS

Spectrum lessons

This issue, Mike Edmunds devotes his attention to Scisoft Educational Software, and offers us his views

I read recently an article in a software magazine which stated that for the potential of the micro to be fully realised then ways must be found of integrating the machine with the rest of the curriculum. This magazine was well over two years old! Yet what do we find now . . . whilst the games player is provided with high resolution multi-screen extravaganzas the child in the classroom is still confronted by drill and practice maths programs or a dressed up version of Hangman (if it happens to be an English lesson!). This view is, of course, overly harsh as there are now many more exciting and stimulating programs available which cover many areas. However, in terms of the total programs used within the educational context it is still very much Maths and English as the norm. The demand for more wide-ranging software seems to be as great as ever. A few software houses are, however, making moves in the right direction. One such is Scisoft, and it is their turn under the spotlight this issue.

Just imagine trekking across wild countryside or striving to protect a rare bug from the unimaginable horrors of the Mystery Mansion, and all from the relative safety of your classroom or favourite armchair at home. These are just some of the options available using the wide range of educational programs for pupils, teachers or parents from Scisoft. A comprehensive list of titles is available covering most subject areas and most age groups from Infants right through to Sixth Formers studying for examinations.

Each of the programs within the range have been designed for use either at school or at home and as such each comes with an extensive booklet to provide instructions and backup materials or ideas for further study. In the packages for the older age ranges suggested activities are included which will help the pupil make the best use of the program. For the packages intended for pupils studying for examination a considerble amount of information has been included within the booklets and consequently they seem an ideal way of home study or revision yet still provide enough material for use within school.

Star Reader

Star Reader is the first of the programs under review and is intended for children of 6 to 11 years. The program, as the title suggests, is to help your child develop his or her reading skills. This tape is Pack B which takes as its subjects Sea and Pyramids. There are three difficulty levels starting with simple sentence construction and extending to more complex sentences with harder vocabulary. Each ability level has two options, Deletion and Sequencing. Deletion requires the child to predict which word is appropriate to the meaning of a sentence. This is a valuable skill but the child has only one chance to get it correct and it might have been more beneficial to allow several attempts for those not quite so able. The child can call up an associated picture but this is rather slow to form and is of rather limited value. It does, however add interest to a mainly text format. The Sea option a similar use of the pictorial element also invites the pupil to write for themselves and the passages are stimulating enough to provide good starting points for storywork. The Sequencing section gives an opportunity for work on alphabetical order or encyclopedia work. There appear to be one or two slight quirks in this section which spoil an otherwise competent program. For example, clues to incorrect responses are highlighted in dark blue, which also happens to be the same colour as all the rest of the text! The order of words is also vital . . . the program accepts '. . . trees and water' but not '. . . water and trees'. As a reading program it has some good points but to be of real benefit would need quite a lot of work with a Thesaurus and pencil, one of the suggestions made in the accompanying booklet.

Even Scisoft have their Maths programs, here are four differing packages covering both Primary and Secondary ages. The first is Maths Plus which covers Tesselation, Number Machines, Graph work, Number Line and a simple business simulation; Factory. Although this is labelled Intermediate level it is certainly within the scope of Junior pupils. The accompanying booklet gives good instructions but, in this case, no additional activities. The tesselation section unfortunately only deals with shapes generated by the nine pentominoes but nevertheless is extremely comprehensive and with the colour, rotation and reversal options can be used to produce many complex designs. The remainder of the programs are equally good and make good use of the computer in an interactive fashion. Again no opportunities for second chances and this is perhaps their biggest drawback.

Maths 1 & 2

Maths 1 claims to be 'exciting and stimulating programs for 9-13 year olds.' Whilst not disagreeing totally I venture to suggest that some of the programs may equally well be achieved with pencil and paper. As home extension however they do provide good value. The program on Fractions however, is very well done and encourages the estimation of differing fractions, with the computer taking a teaching role in the event of a mistake. Other programs on this tape are Symmetry, (using line or rotation — this program also has a printer option to give hard copy of the design) Shape (to name given mathematical shapes), Battleships and Factors. This last program requires the necessary factors of a given number. The big concern here is the fact that this section is SLOW!

Maths 2 also has five programs. 'Lemonade' is the type of program that has been around for a while now and the idea is to make your fortune selling lemonade under various condi-





tions. 'Angle' requires the pupil to estimate a drawn angle. The next program 'Glass' is a simplified version of Vu-3D and as such will aid the design of different glass shapes and allow them to be manipulated through different angles. Just like Vu the frameworks are of the 'wire' type but are very effective. There is an option to change the resolution of the drawing but unfortunately this contains a BUG which spoils an otherwise excellent program. (Note to Scisoft, change an A$ in line 2150!) 'Percentage' is an effective estimation program encouraging the user with humourous comments such as "That was a pretty wild guess . . .". Overall therefore, a fairly good mixture on this tape.

Maths 2(13+) consists of six mathematical programs and a Revision section. This program is intended for examination students who are expected to have thoroughly revised the areas concerned. These areas include all aspects of trig. ratios, sine, cosine and tangents. Pythagoras Theorem, percentage change, proportions and scales. Each of the programs have up to ten problem types and within each program random numbers are generated to give a wide variety. There are no second chances at a problem, indeed you are expected to find why you have made an error. At this level this is no real detriment to the program. Your attempts are given a score to give an indication of your competence in any area. All of the programs supply good examples and, being almost purely mathematical, generalised comments on each are not really relevant. The booklet is helpful and outlines all the necessary rules contained within the programs. The triangle diagram here is slightly suspect however!

Intermediate Level

Moving away from Maths we come to Scisoft's Intermediate level programs, the first of which is **German 9-13**. This, in essence, is almost exactly like 'Star Reader', the main difference being that all the required inputs and sentence work is, of course, written in German! An overlay is provided to give access to the required accented letters and other symbols peculiar to the language. I must admit that my German is decidely tourist type, but even so I got some right. The Help facility provides an English translation of the required word and will also give a complete translation at the end for checking. The beauty of this program is the interactive aspect, not always immediately available in a crowded classroom. It is claimed that these are not simply vocabulary tests and whilst they actually appear to be JUST that, there is the opportunity to put the program to a much fuller use in terms of translation work etc. 'Alphabetical Order' requires translation before sequencing, and 'Encyclopedia' needs to determine where an English equivalent would be found.

Geography is a less brainteasing program and decidely more fun! Four programs this time; Treasure hunt, Around Britain, Spread of disease in woodland and Countries. Treasure Hunt contains three games based on map reading, use of compass directions and estimation of distances. Each game is played on a choice of eight maps which use standard Ordinance Survey symbols. This is a thoroughly enjoyable set of games and successfully reinforces the concepts concerned. Around Britain requires careful plannng to achieve success, the aim being to travel around in search of hidden letters which spell out the name of a city or town. There are numerous options and it is up to the individual to make the best use of available resources. there are also two short sections on place names and the estimation of distances between cities. (This last section should be tackled warily as the distance answers do not conform to any of the maps that I consulted!) Disease is a variation of the old standard 'Life' except that it details the spread of disease within woodland. 'Countries' is by far the most impressive of the programs in this package and provides a comprehensive database on the major countries of the world. The database contains various details such as capitals, areas, population etc. A wide range of conditions is available as search options. this set of programs is accompanied by an explicit manual which also contains additional activities for the user to try.

The next program is **Computer Studies** and is intended for pupils in the 13-19 years age range. There are seven programs dealing with flowcharts, an introduction to machine code, logic gates, the application of gates, truth tables, a logic processor and number bases. The programs are of the question and answer type and mainly text although clear diagrams are used where necessary. I was impressed by the straightforward nature of these programs and they should be of real benefit for the target students. These would also be of interest to anyone wishing to go beyond the BASICs in this area! CES for example gives a fairly clear introduction to machine code, and even I could follow with no trouble at all. There is also a revision program and mock exam included . . . more of which in a minute.

The Appliance Of . . .

Science 1 is a novel type of program but I am afraid it is not up to the standard of the other programs! 'Bug' is intended to be an adventure game for children, the aim being to find a box hidden within Mystery Mansion. This box contains the necessary requirements to keep a rare South American bug alive. Travelling through the mansion you are confronted by various problems which require scientific answers in order to proceed. The framework is fairly standard adventure type but there are too many minor holdups and inconsistencies to let the program flow smoothly. For example, in one room you are required to manufacture oxygen from given substances . . . however if you make something lethal instead you do not appear to die immediately. A performance report and the status of the Bug is given at the end. In all fairness, although this program did not appeal to me it nevertheless is a different approach to Science. There are also two database programs on Elements and Organisms which both succeed in their objective without being really anything out of the ordinary. 'Plant' follows the same style as Bug, the aim this time being the pollination of the plant. Again this did not really succeed for me. The answer here is that if this sounds to be what you want then try it first.

The final two programs I intend to deal with together as they are essentially the same format. Both the **Chemistry** and **Physics** packs come complete with a very comprehensive, fully illustrated booklet which is intended to help with the students revision programme. Both programs have problem sections which are supplemented with graphics. Also included are several programs which include multiple choice questions based upon equations and definitions of relevant terms. Physics covers areas such as Mechanics, Electricity, Heat, Pressure, Light and Sound whilst the Chemistry program covers Acids and Alkilis, Mass/Mole calculations, Gases, Metals, Elements, Ions and Electrolysis problems. A review such as this cannot hope to deal with such widely based programs with any degree of detail, suffice it to say that I would unreservedly recommend these packages as revision aids where it not for the Revision sections! . . . These, in my opinion spoil the whole package. Whilst the suggestions for programmes of revision are helpful and concise I see no need at all for the attitude that is put across! The concern over failure is surely real enough for any student without having fail certificates flashed onto the screen. The comments for this advice section also strike me as unnecessary, even if they ARE deserved "Stop kidding yourself, you are almost an adult now!", "If this is the first time you have thought about revising then we suggest you try prayer".

In conclusion, the Revision sections aside, I see these packages as being of considerable value, especially as extension packages for use within the home. Each, within the limitations imposed by the nature of the program, makes fair use of colour and sound and the excellent booklets add an extra level to the overall presentation. However, even given such comprehensive packages as these, and bearing in mind that they tread fairly new curriculum areas, I still maintain that there is still a long way to go before the micro realises its full potential within the school timetable.

So come on Software houses, put some of the profits from your Megagames into providing programs of equal quality for the Educational sector!

BREAKTHROUGH!

SPECTRUM DISK INTERFACE

The Ultimate for the Spectrum owner!

Spectrum owners! Now you can upgrade your machine to take *full advantage* of the sheer speed and convenience of floppy disk operating.

Greater storage! Faster access! More operating flexibility!

Disk drives were specifically designed for use with computers so programs and data can be stored and accessed quickly and safely. Also a disk drive uses random access to get programs and data from any part of the disk. The speed? Only a few seconds!

Upgrade *now* with this low-cost Beta disk interface from Technology Research that uses only 128 bytes of Spectrum memory and gives you *instant* operating compatibility with any disk drive that the BBC micro uses!

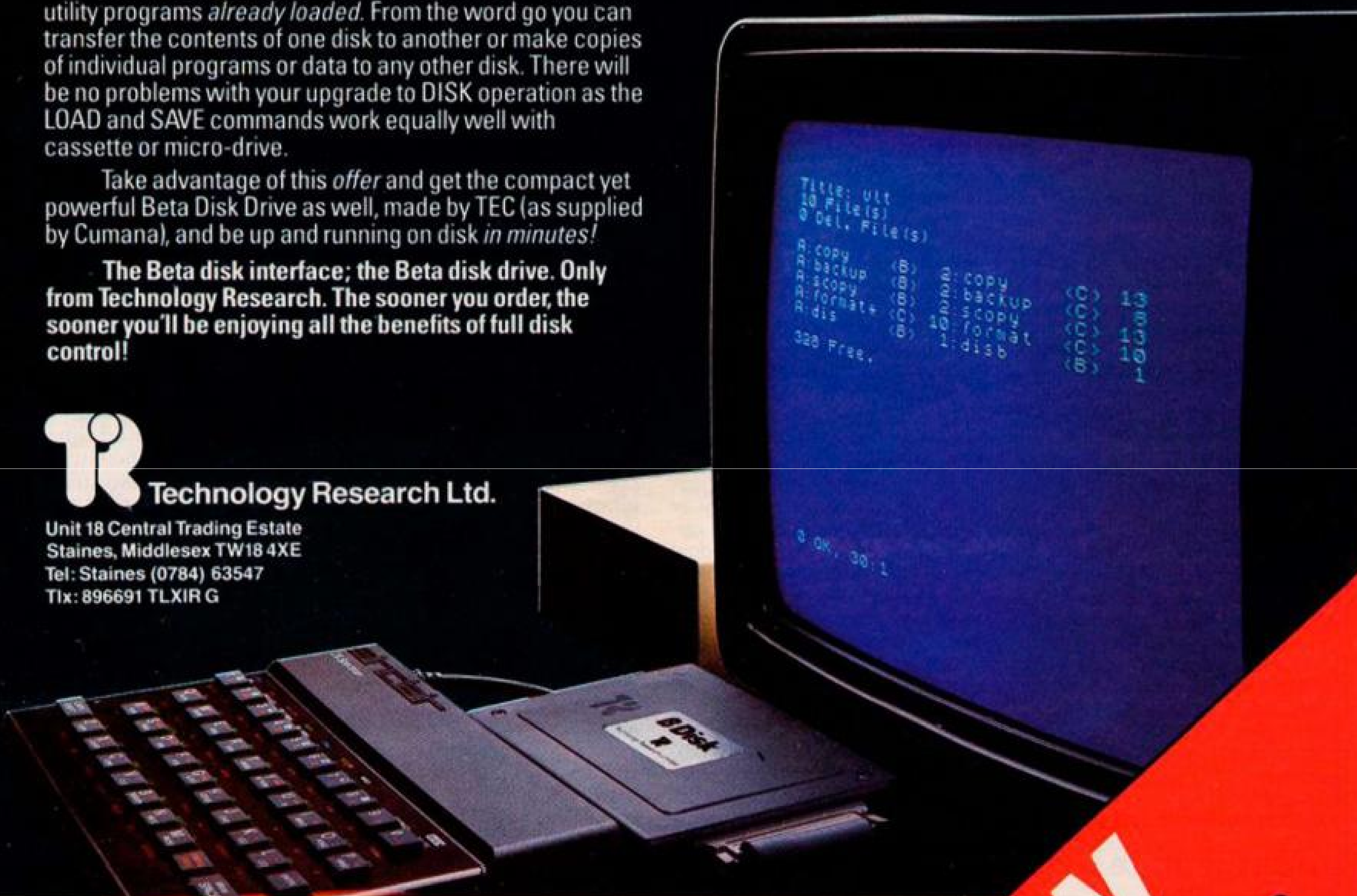
Each interface comes complete with a 5¼" disk with utility programs *already loaded.* From the word go you can transfer the contents of one disk to another or make copies of individual programs or data to any other disk. There will be no problems with your upgrade to DISK operation as the LOAD and SAVE commands work equally well with cassette or micro-drive.

Take advantage of this *offer* and get the compact yet powerful Beta Disk Drive as well, made by TEC (as supplied by Cumana), and be up and running on disk *in minutes!*

The Beta disk interface; the Beta disk drive. Only from Technology Research. The sooner you order, the sooner you'll be enjoying all the benefits of full disk control!

Features

- EPROM disk operating system for ease of operation
- Available in Single Density and Double Density
- Uses only 128 bytes of Spectrum memory
- Compatible with 40 or 80 tracks, single or double-sided disk drives
- Works equally well on 5¼", 3½" or 3" disks, with upgradability built-in from 100K to 2.6 Mbyte
- Supports up to four disk drives
- BASIC program merge facility
- Duplicate Spectrum connector included
- Uses Spectrum key words
- Random access of data for fast operation
- It costs **less** per byte of storage to switch from tape to disk



Technology Research Ltd.
Unit 18 Central Trading Estate
Staines, Middlesex TW18 4XE
Tel: Staines (0784) 63547
Tlx: 896691 TLXIR G

CUT OUT THIS COUPON AND SEND WITH CHEQUE OR POSTAL ORDER TO THE ADDRESS ABOVE

PLEASE SEND ME:		QTY		TOTAL
SINGLE DENSITY BETA INTERFACE	@ £ 97.75		£2 p.p. (UK)	£
DOUBLE DENSITY BETA INTERFACE	@ £109.25		£2 p.p. (UK)	£
S501 BETA DRIVE*	@ £152.00		£6 p.p. (UK)	£
				£

*40 track single sided, including power supply
Prices for other disk drives available on request

Name

Address

Postcode

Allow 28 days for delivery

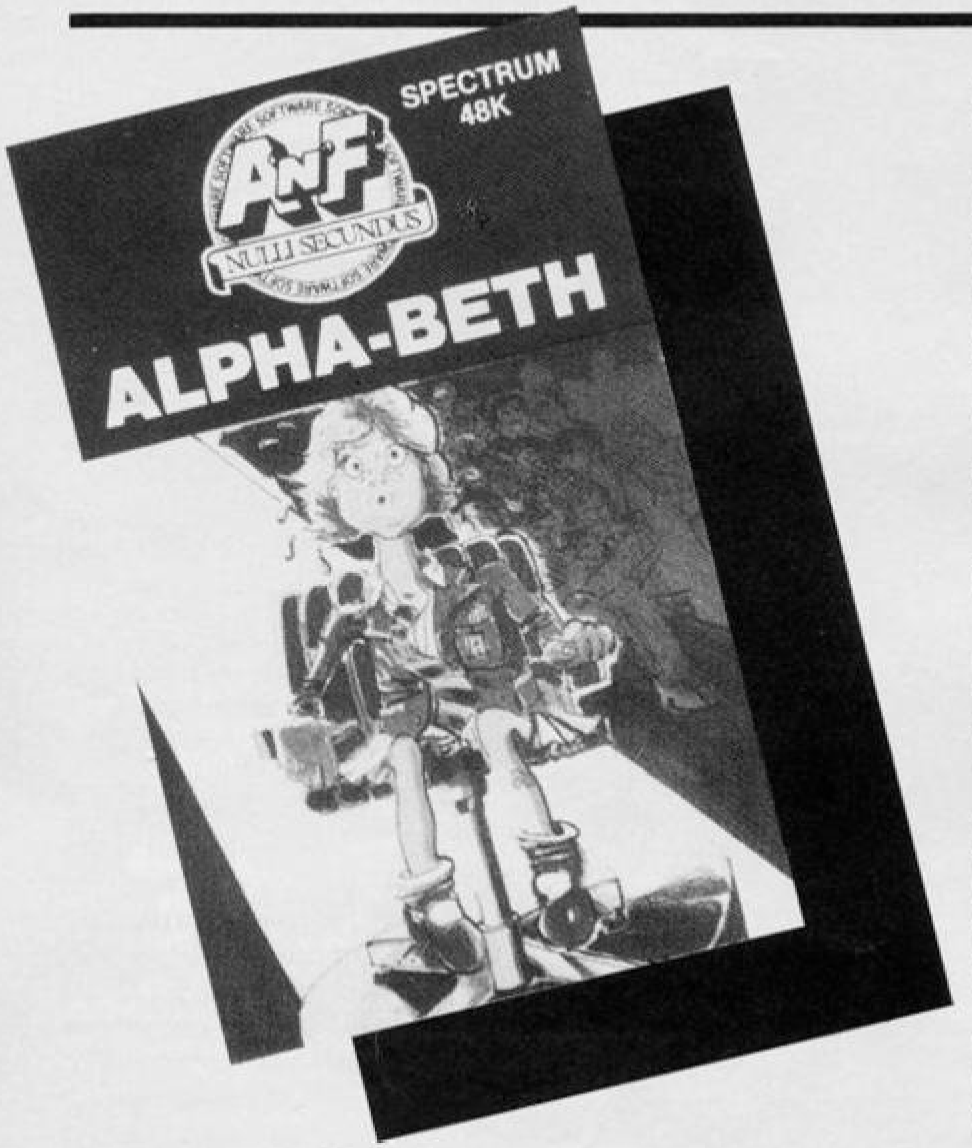
NEW

BETA INTERFACE PLUS BETA DRIVE!

The complete package from £249.75!

Quicksoft

Clive Smith continues his quest and unveils another mixed bag of programs recently released



Alpha-Beth by A'n'F

Written for the 48K Spectrum, "Alpha-beth" is an educational/arcade game which is designed to make learning fun.

Basically it's a quiz game but answering the questions is quite novel.

You are shrunk down to the size of a pea (in fact you are a little schoolgirl) and faced with a giant keyboard and VDU. The questions are printed across the VDU and you have to answer them by leaping from key to key pressing the right letters to form your answer. Once you have printed out your answer you have to make a mad dash for the "Enter" key to see if it's correct.

Things are not all rosy though, as you are chased around the keyboard by the schoolbully and an object which changes in relation to the questions. If you bump into one you lose one of your six lives.

I had trouble loading as the inlay said LOAD " " and it wouldn't load until I entered LOAD" "CODE. Once loaded you are presented with a menu and on asking for instructions I received about three pages in half a second.

I rang the company to ask if my copy was a 'special' just for reviewers but they confirmed it was a production model.

Other options on the menu included one to save and load your scores from previous games. Another option enables you to select which operating keys you would like to choose or another to use a Kempton joystick.

There are several girls names listed with scores of 1000 by each one but not being able to read the instructions I was quite baffled by them.

Once you have answered all the questions you can go out and buy 'ALPHA-BETH (gen)' with a database programme so you can write your own questions. Also there are to be prerecorded data bases on varioussubjects to be released. These can be loaded from the menu.

The game itself, I would imagine, is aimed at the 8 to 15 year olds.

Other games from A & F are Jungle Fever, Cylon Attack and Chuckie Egg.

Projector 1 by McGraw-Hill Book Company

An ideal program for use in schools, business training or those dreaded sales meetings. Projector 1 enables you to build and present graphs, histograms and pie charts easily and quickly.

It is treated very much like a projector but instead of making up slides you now use a TV screen and store your screens in memory to make up a presentation.

On loading the program you are asked for information on the type of interface and printer you are intending to use and from a menu you choose the option to match.

There is a comprehensive demo program already implanted to show you it's capabilities as well as a 15 page instruction book to help you.

After running through the demo you have to clear all the data and start to build up your own presentation.

You start at a main menu and select 'New data'. This gives you a range of formats to choose from ie: Years, months, days etc. Following screen prompts you are asked to enter your values, I entered sales on a monthly basis for one year. Once all your data has been entered you have to then give it a name, say 'sales'. This is then stored and is listed in a directory for use later.

Now returning to the main menu you can start to build up your presentation. From the data entered you can choose one of the options to either produce a pie graph, histogram or graph.

I started by pressing the graph option which brings you to a sub menu. All you have to do here is choose what calibrations, colour, titles you want by a series of yes (Y) or no(n) answers. You have the option to connect up your points, have a background grid or put in a fit line or curve.

Once completed the grid is automatically given a reference number and as it's the first graph it's called G1. This is also stored in the directory.

Building the histogram and pie chart is done in much the same way as the graph and once done these are also given a reference number ie: H1 & P1.

With the histogram you can have an overlay, say overlaying last years sales figures with this years. Another feature is the programs ability to forecast or project up to another six places. Now all these graphs etc can be linked with text, this is limited to headings and sub headings but should be adequate for most needs. All the text is written out in enlarged characters so it can be easily read. Each page of text is also given a reference number (K1) and is also stored in the directory.

Going back to the main menu you now choose the directory and you can view all the graphs etc listed.

Now it's 'fun time' as you now press the 'build/modify-presentation' option from the menu. You then go to a sub menu and choosing 'build' you can put the presentation together.

This is done by listing the reference numbers in the sequence you would like them to appear. So, first you would probably start with text, so you enter K1. Then maybe a graph to follow, so enter G1. After the graph perhaps a couple more pages of text, enter K2 & K3 and to follow that a histogram, enter H1 and maybe a pie chart, P1.

All the entries can be put into any order you wish and each page can be used several times.

Once the presentation is complete you have the option of letting it run automatically or manually so to fit in with speech. If you choose the automatic option you can set the speed

you want each slide to appear. If done manually you have control via 3 keys to replay the slide that has been shown or to return to the previous slide or to make a hard copy if needed.

Once the presentation is running you can also break and stop it which takes you back to the menu.

With the presentation complete you are then asked to give it a name and that too is stored in the directory. It's possible to link presentations together to form a larger programme.

One of the things this program lacks is the ability to draw graphics. However all is not lost as it does give you the ability to load a pre-defined screen image which could have been drawn on one of the painting programs now available. This could be a logo for your company which will give your presentation a more professional finish.

The only gripe I had, was in the graphs. The calibrations that are written on the vertical side are written inside the area which the histogram is printed. This makes them unreadable when the histogram is quite full.

Overall I think this is a useful tool to have for any company or training school.

The Highway Code by Datek

Take ten house points and gold star Datek, for the most helpful program I've ever come across.

If you are in the throes of learning to drive then you must have come across the problems that greet every learner, finding someone to test you on the highway code. Look no further, here is an excellent program that will take care of your troubles.

Once loaded you are greeted with a menu and you are asked to stop the tape. Then you are asked if you want the first set of questions. All the questions come in sets. The first set is on motorways, then signs that give positive orders, then warning signs, general signs and finally signs giving negative orders.

If you want motorways you press 1 and start the tape. This will load the questions and present you with a second menu.

Each question comes with a very well drawn road sign and you are asked from a choice of several answers to pick the correct one.

Via the menu you can use the program in three different ways. Firstly as a learning aid, where you are told what the signs are, or as a quiz where you answer against the clock and go back through your mistakes and be told the correct answer. Or thirdly, answer the questions and be shown your mistakes as you go.

The program will keep a score to see how you progress and you also have the ability to make a print out of either scores or questions.

On the flip side of the tape are two more short programs. The first tests you on traffic light sequences with the aid of some well drawn graphics.

The other program is a demonstration of a level crossing using some very clever graphics and really is fun to watch.

At this moment in time I don't have a highway code book so I cannot check to see if it contains all the road signs. One thing I didn't see was braking distances, but it does hold most, if not all the road signs, parking restrictions and road lines.

If you are taking your test shortly I can really recommend this to you, oh! and best of luck with the test.

Micro Arts from Micro Arts

Now I'm always looking out for the unusual programs and here is something that falls into that category. It's where art meets the computer.

Micro Arts is a forum organisation and will accept your pet projects for publication. They run on a very low budget (does this mean you don't get paid?)

I was given two tapes to look at and on loading the first you are presented with a menu under the heading 'Animated Originals'.

These animated originals are different shapes and colour that move around the screen. The colour, timing and order can be changed via a menu.

I have to truly admit I have seen better graphics before, I think it was when I had a program crash once. I like my art with oils, personally I think this is whitewash.

Still, people must have thought Picasso was a bit on the weird side when he started.

The second tape was also menu driven, this gave a choice of 6 different graphic displays much the same as the first tape.

One of the options I pressed gave me a graffiti type man so detailed that he even had his little credentials showing, and what he does I dare not say.

The first tape will set you back £3.00 inc P&P and you also get a magazine. There's even a piece from Clive Sinclair in it. The Second tape is £2.50 inc P&P and magazine.

Well if you like get into computer art or want a copy of one of these tapes then write to: Micro Arts, PO Box 587, London.



Zenji by Activision

When working with computers for some time you tend to end up like Spock, everything you do is done by logic. Well here is a game where you have to use your intuition instead.

Zenji is a maze of passages, but the maze is broken into separate pieces. The idea of the game is to connect the pieces and complete the maze.

This is done by moving a cursor from piece to piece and turning them around till they all connect. Doesn't sound much fun does it? I have to admit when I first played the game I thought it to be a load of rubbish I was getting nowhere.

I was trying to do it logically, so I tried again and went by my instinct and got on really well (could this be the power of Zenji?).

You have a choice of joysticks you can use which is needed as you play this game against the clock. Points can be gained whenever you make a complete connection.

The first maze is small but as you progress the maze increases in size, and in the later stages you are chased about by small flames which burn you up.

Although a simple game I found that I couldn't put it down which is a sign of a good game.

Therbo from Arcade

The game is a bit like space age football. Written with some well drawn graphics you are presented with a wide corridor which runs vertically across the screen. At each end is a goal, one is yours and the other is the computers.

The ball is replaced by a flaming cursor which is fired towards your goal. To stop it there are strategically placed thrusters which fire little thingies across the corridor and you have to hit the cursor and knock it to the side so it blows up.

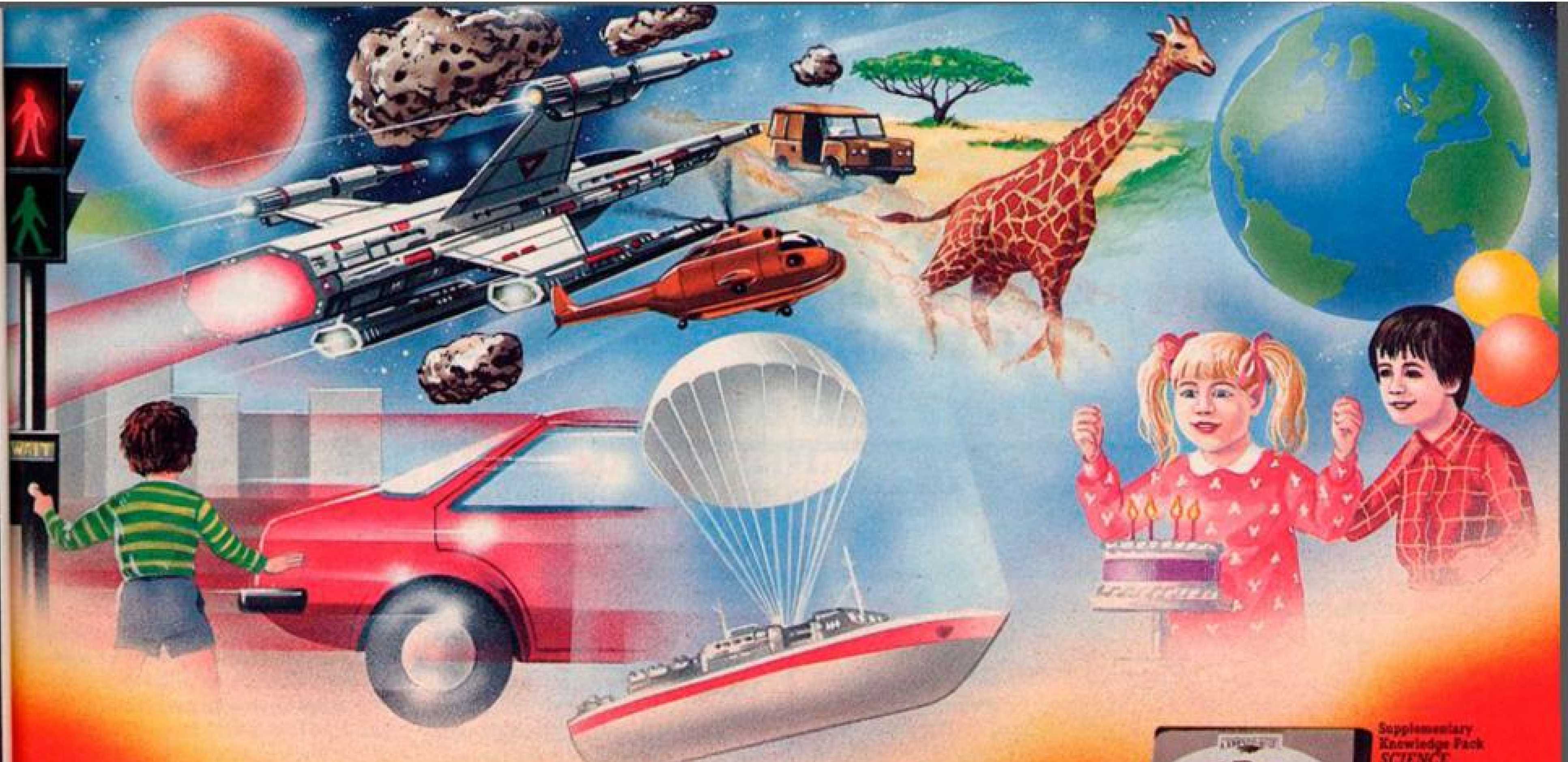
You are playing against the computer and you have turns at each others goal. When you have a go at the computers goal the computer fires thingies at your cursor.

To assist your cursor toward the computers goal you also have a tank which fires missiles. If the computers cursor is nearing your goal and you think you cannot stop it in time you can press the Mesmo button which blows everything up. These can only be used three times though.

Added to this there are polarity missiles which may also be fired at you as they approach you can protect yourself by guessing it's polarity. If it carries a negative charge you can send out a positive charge to blow it up.

There are 13 keys to operate, some which can be handled by a joystick.

To tell you the truth I was not impressed. The graphics are good but the game itself is boring. I even showed it to a friend who also confirmed my views.



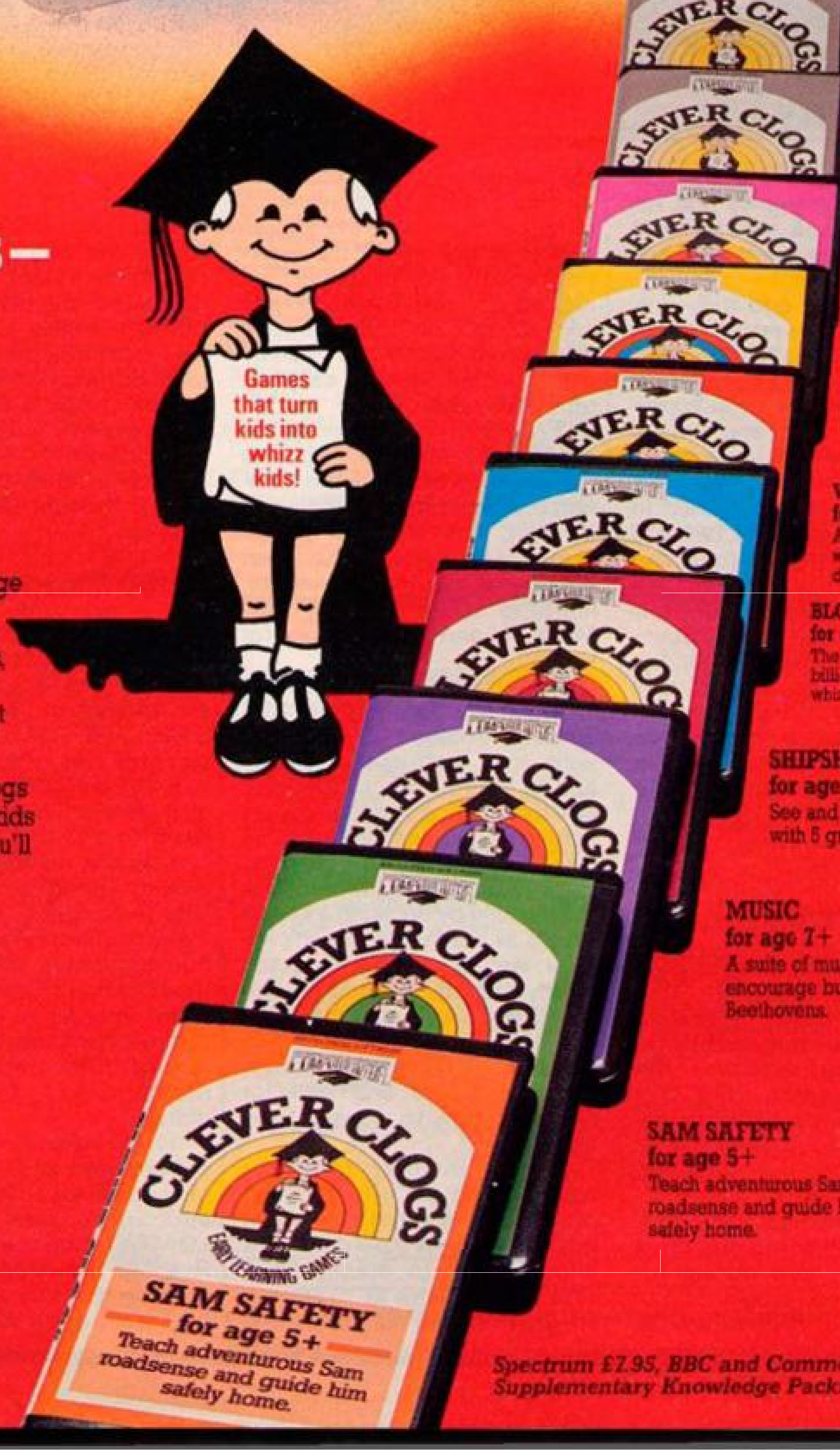
'Calling all whizz kids – my CLEVER CLOGS games are the ultimate challenge!'

These exciting games of skill and chance are the combined work of real life 'Clever Clogs' and their teachers to give you whizz kids the ultimate challenge in computer games that really make learning fun.

Exciting graphics, a wide choice of 'difficulty' levels, plus supplementary knowledge packs for several games in the range make Clever Clogs Britain's best buy in early learning software.

Starting with 'Party Time' for Age 3+, the Clever Clogs range brings a new approach to learning for whizz kids of all ages. And if you join the Clever Clogs Club you'll automatically be eligible for many money-saving special offers.

Selected titles available at W H Smith, Menzies and other leading retailers.



Supplementary Knowledge Pack
SCIENCE
Challenge your whizz kids with 600 science questions

Supplementary Knowledge Pack
THE ARTS
600 first steps into Music, Poetry, Painting, Dance, Theatre and English Literature

STAR TRUCKER
for age 9+
Hyperspace holds no fears for whizzkids who get the answers right.

PARTY TIME
for age 3+
5 different games and increased difficulty options teaching letters and numbers the fun way.

JUNGLE JUMBLE
for age 5+
You've never seen a zoo like this one – a laugh each time with 2 different games and increased difficulty option.

WHIZZ QUIZ
for age 7+
A real challenge to todays whizzkids with increased difficulty option.

BLOCKBUSTER
for age 7+
The puzzle with a million billion variations. Can your whizz kid find the answer?

SHIPSHAPES
for age 3+
See and understand SHAPES with 5 great games.

MUSIC
for age 7+
A suite of musical games to encourage budding Beethovens.

SAM SAFETY
for age 5+
Teach adventurous Sam roadsense and guide him safely home.

CLEVER CLOGS
early learning games

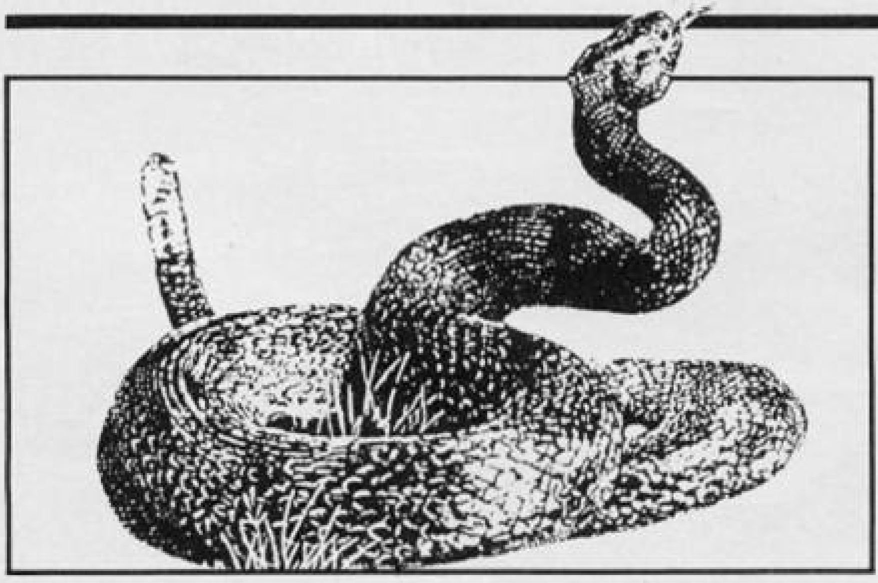
produced by
ARGUS PRESS SOFTWARE
1 Golden Square, London W1R 3AB
Telephone: 01 437 0626

Argus Press Software Group

Spectrum £7.95, BBC and Commodore 64 £8.95, Supplementary Knowledge Packs £5.95.

Thurnell Disc Drive

An exciting machine looked at by the ZXC team



I recently met Chris Drinsmead of Thurnell Electronics and we spent some time talking about their 3" drive unit and the market in general.

A little while ago they produced the first version which sold well, especially in Europe, this was then followed with unit MK2 and has now been superceded by the latest and what is likely to be the final version.

Chris explained, "We have used the experience gained in the making of the other two and the feedback from customers to produce the machine that we believe the market is looking for. The latest version uses no Spectrum memory whatever and is still easy to use, has Hitachi reliability and 300K capacity" Well, I decided to call on the opinions of a general user rather than a hardware expert to put it through it's paces, so over to Tony Woods.

The Drives

In the time that it takes to blink an eye, the Thurnall Disk System will LOAD "Tasword Two" to enable this reviewer to start to type, whereas the lengthy cassette LOADing method enforces a coffee break. It has a similarity to coffee in that its SAVing and LOADing are INSTANT.

The well-packed outfit consists of Hitachi based disk drive power supply, user manual for disk basic and programmers' reference card. With power to the computer and all "add-ons" switched "off", the drive can be plugged into the expansion socket at the rear of the Spectrum and the power supply connected to the mains. Upon switching "on" the legend "Spectrum disk BASIC" can be used, including the cassette tape commands.

The "Welcome" card, after explaining the above, suggests the insertion of the demonstration disk and pressing the black button. Instantly a menu appears offering the four options of playing Snakebyte, learning about disk BASIC, MERGing and zero memory for EXITing.

The game Snakebyte allows Thurnall to demonstrate the speed of access of the system, with four instruction screens and an offer to choose a level of difficulty preceding the games screen. Learning about disk BASIC from the nine screens, some with four extra sections each added by pressing a key, is simple as these instructions are clear, well-designed and not too wordy. It firstly explains how to LOAD and SAVE, and then how to save and load television pictures, numeric arrays, strings and string arrays, and finally for Code and how to reassign new addresses.

I was now eager to try my new toy and set off, clutching my box of goodies, to pay a quick visit to my local computer expert who, in double-quick time, altered lines in "Tasword 2" to enable it to LOAD and SAVE on disk. He made it appear so easy and explained how it was done, line for line — so that I could adapt other programs. When I reached the safety of my own computer and reconnected all the equipment, I could not remember how to use all the POKEs and MERGEs that he had mentioned!

Thus it was that I was forced to approach the "User Manual", which fills in the gaps left by the "Programmers' reference card" and the introductory program. I would have liked to see a larger format manual than half A5, the use of larger type than 8pt and an appreciation of the problems experienced by some of us slow learners. So I donned my spectacles and proceeded to learn that the thin black button ejects disks, that the black button on the left makes the computer load from disk and is called the reset button, and that the commands in disk BASIC resemble their cassette counterparts with the addition of "AND FN d ()". This is going to be a piece of cake, I thought to myself, and read on.

Auto Execution

Next, I was introduced to making programs self-executing after they have been loaded, by the addition of "LINE n" to the SAVE and LOAD statements. Not only was I then shown how to SAVE and LOAD pictures, but also given a five line picture program to try, and — it worked, just as the manual said that it would. There, in the directory, was the evidence in the form of "picture screens$". On to numeric arrays, with a possible modification to the LOAD command in order to load only the desired numbers by the inclusion of "DIM a(n);" before the normal DATA loading command.

I fairly sailed through strings, string arrays and code to arrive at page ten breathless, to find instructions on how to SAVE and LOAD my own startup program. Rubbing my hands with glee, thinking that this was more like it — really using the disk unit at last, I entered the two line program and "SAVE "USR" AND FN d() LINE 10" and then pressed the black button. There on the screen were the words that I had typed in! This was followed by the LIST command that prints out the names of what is stored in each of the forty-four portions of side A of the disk, each one representing 3.25 kilobytes of data.

How to ERASE a file came next, complete with a means of CLEARing a whole disk, which is something that must be done to all new ones. Then there were instructions on how to transfer a program from cassette to disk by first using the MERGE command, giving access to disk BASIC, and then the usual SAVE command.

I was now at page sixteen and had not become lost in any of the above, as the descriptions were very clear and concise. Here were three pages on reports and error messages that appear at screen bottom, giving their meanings and the situations when they occur. The last six pages consisted of an explanation of how to add the access to disk BASIC to programs in other places than line one, how to have random access disk files in disk BASIC and some technical data for the more experienced user.

But, sad to relate, nowhere are there instructions on how to SAVE all those programs that I have bought and take three to four minutes to LOAD from cassette. Why have such a superb system at our disposal, giving instant retrieval of information and programs, if they can only be used with "home-made" programs? Do software houses really believe that every Tom, Dick and Harry are going to rush out and invest vast quantities of cash on a disk drive in order to copy their friends games and the like? As most games players are children or teenagers it would seem unlikely. Having purchased a games program it would be ideal to be able to put it onto disk for instant LOADing, without having robbed the manufacturers of their cash. I should think reviewers would find it a boon!

Interestingly Thurnell are negotiating with a software house with a view to producing a "tape to disk" program, which should be worth waiting for, and they can already supply Tasword II and Masterfile in disk versions if required.

The complete units costs £219.95 and is only available from Thurnell directly at the moment.

Thurnell Disc Drive Competition

Your chance to win a superb disc system for your Spectrum

Be a proud owner of one of these super systems, the latest in the range of professional add ons for the Spectrum which bring it in line with machines costing much more.

All you have to do is study the list of benefits that the disk drive system provides and put them in what you consider to be the order of importance.

Chris Brinsmead of Thurnell Electronics will supply us with his order and the entry which matches or comes nearest, will win one of their units.

In case more than one entry matches Chris' order we ask you to complete the sentence on the entry form in ten words or less. Should we have a tie break situation then we'll ask him to decide which he feels is the most appropriate sentence.

So, thinking caps on, pencils at the ready and away we go!

- A fast access
- B reliable storage
- C good backup from Thurnell
- D easy to use
- E clear instructions
- F low cost
- G good looks
- H compatibiliti with other peripherals
- I 300K capacity d/sided drive

On the entry form fill in the letter of your first priority in box number 1, the second in box 2 and so on. Complete the sentence which begins: "I would like a Thurnell Disk Drive because . . ." In not more than ten words, cut out the form, put it in an envelope (don't forget the stamp) and pop it in the post box. Now all you have to do is sit back and keep your fingers crossed.

The Rules

- The competition is open to all UK and Northern Ireland readers of *ZX Computing*, except employees of Argus Specialist Publications Ltd. Their printers and distributors, employees of Thurnell Electronics, or anyone associated with the competition.
- As long as the correct coupon is used there is no limit to the number of entries from each individual.
- All entries must be postmarked before January 31st 1985. The prizes will be awarded to the first ten entries picked at random which bear the correct answers, the decision to be made by the Editor of *ZX Computing*. No correspondence will be entered into with regard to the results and it is a condition of entry that the Editor's decision is accepted as final.
- The winners will be notified by post and the results will be published in a future issue of *ZX Computing*.

NAME . . . .

ADDRESS . . . .

. . . .

. . . .

"I think the important features of a disc drive should be in the following order:

1
2
3
4
5
6
7
8
9

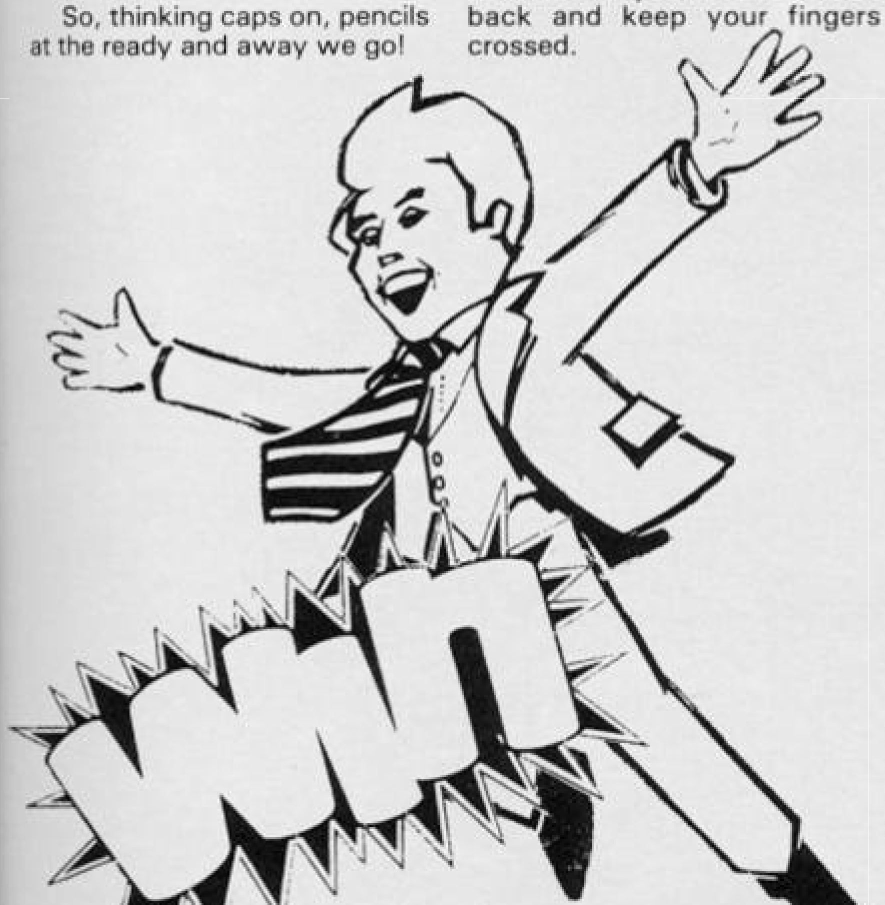
"I would like a Thurnell disc drive because:

. . . .

. . . .

. . . .

. . . .



Prism Competition Results

The October/November PRISM competition prompted a massive response and most of those readers who entered identified the five photographs correctly. The pictures (in order from one to five) were:

1 ZX81 Keyboard
2 ZX Printer
3 ZX Interface 1
4 ZX Microdrive
5 ZX Spectrum

A regular piece of cake, huh? And so, the ten lucky winners picked out of the editors wide-brimmed stetson are:

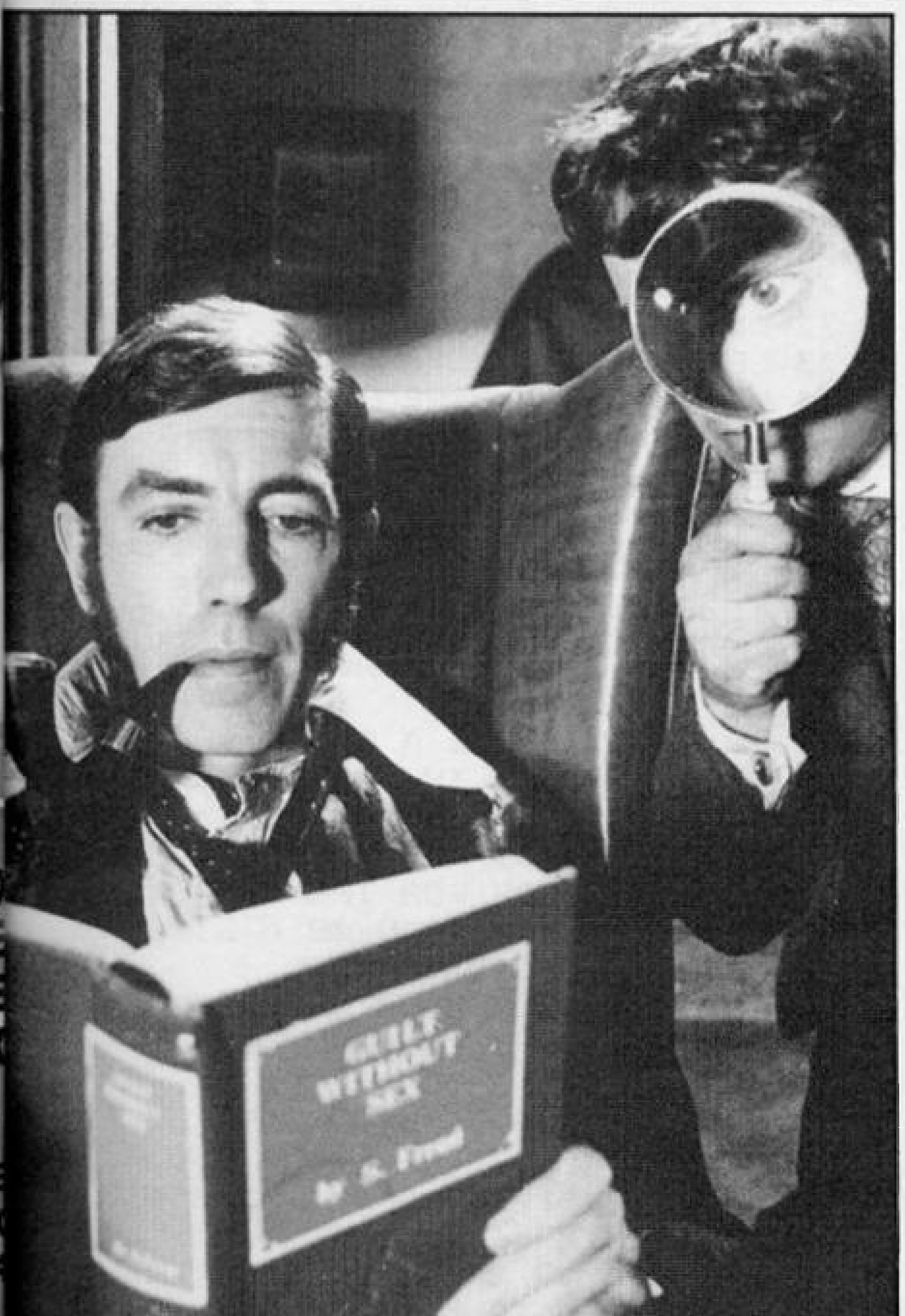
Mr Julian Payler of Northampton
Mr A Ford of Cornwall
Zamir Hyder of Surrey
P J Woodward of Bradford
Barry Hetherington of Thame
Daniel Turtschi of Switzerland
Mr A D Mallet of Bracknell
Mark Summers of Northumbland
N A Lytton of London
and Bobby Watson of Ayrshire

Congratulations to all those, and my deepest sympathy to the unfortunates, but thanks to all those who entered.


MIND GAMES
AMERICAN FOOTBALL
A fast & furious graphics strategic simulation
£9.9
FRE
RUL
BOO
FOR TH
SPECTRUM 48
CBM 6
OTHER GREAT MIND GAMES:
MIND GAMES QUEST FOR ETERNITY
Starring The Overlords of the Universe
The candidate (you) have to get to the Chamber of Creation. It's a laugh a minute, since it's 2,000 light years away on the most horrible planet in the Universe ... and your starship doesn't work either!
MIND GAMES ORION QUEST
A full feature adventure starring well known nasty aliens the Zarps. Can you play the hero and stop their plans to blow up the earth.
MIND GAMES STAR FORCE SEVEN
Starring The Zurgs
After a desperate spa
battle only one fleet
heroes remain to
prevent the invasion
earth. The future of
humanity lies with y
Argus Press Software Group
For mail order, write with cheque/P.O./card No. to:
Mind Games, Argus Press Software Group, No.1 Golden Square, London W1A

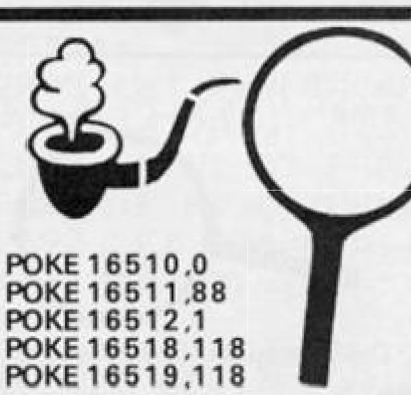
Oodunnit?

Sherbet Homes would have had a problem solving S Greaves' mind stretching game which was set in Coventry.



OODUNNIT is based on a popular board game and is written for the 16k ZX-81. The game is played against the computer. To win the game you must find out "who done it", where, and what with. But first let me explain how to enter the program.

At first sight listing 1 may look like machine code, but it's not. It is, in fact, data for the computer to move around the screen without running into the walls. The data is stored in a REM statement 342 characters long. To make this first enter line 1 REM 110 characters (any characters will do). Next edit line 1 to make 3 lines (1 REM, 2 REM and 3 REM). Check that PEEK 16856 = 118 if not check the length of your REM statements. When all is correct enter the following as direct commands:-



```
POKE 16510,0
POKE 16511,88
POKE 16512,1
POKE 16518,118
POKE 16519,118
```

If you now list the program you will find lines 1, 2 and 3 have vanished and line 0 REM has appeared, if not you will have to start again. When you have got your REM statement enter the main program listing 2 (lines 9700 onwards of the program is the data loader and reader). After entering line 1 RAND the computer should list it, if not enter as direct commands LIST 1 N/L, POKE 16419,1 N/L. When you have entered the program you have to enter the data. To do this RUN 9700 and, when asked to enter the starting address, enter the address at the start of the first line in listing 1 (16514), then type in as many bits of data as you like followed by Newline (please note one bit of data is 2 characters long). Do not enter a space between data numbers. If you make a mistake, RE-RUN 9700 and enter the address at the start of the line you made the mistake in, and re-enter the whole line again.

When you have entered all the data you can try running the game. On running the program you should first get the instructions and be asked to press any key to continue. After pressing a key the screen should go blank for a few seconds before a list of your cards appear (3 Rooms, 3 Weapons and 3 Names). Pressing any key will then start the game and you should see a plan of the house. When told to make your move use the cursor keys for left, right, up and down and keys 1, 2, 3 and 4 for diagonal movement. If you move onto an inverse number in a room you will be asked to make a guess, the computer will then tell you if it has any of the cards you have guessed. After your guess you may use up any moves you have left. When your go is over the computer will make its move and, if in a room, make a guess and ask you if you have any of the cards it has guessed (the computer does not access your cards in making its guess, but if you tell it you have a particular card it will remember it).

The game continues until you or the computer guess right. All being well the program will run OK first time, if not here is a break-down of the program to help you find the bugs.

Program Breakdown



Lines 10 to 99	Set up the variables and print the instructions.
Lines 100 to 300	Deal the cards and then start the game off.
Lines 1000 to 1199	Print the floor plan and the players.
Lines 2000 to 2200	Check the keys, move your piece and also checks you have not run into a wall.
Lines 3000 to 3999	Check whether you have hit a wall or are in a room, lets you make a guess then checks if it is right, or tells you if the computer has any of the cards you picked.
Lines 4000 to 4500	Moves the computers piece.
Lines 4500 to 5999	Sorts out which room the computer wants to go to and which data to use.
Lines 6000 to 7000	Make the computers guess.
Lines 9000 to 9320	Are various subroutines.
Lines 9400 to 9540	End of game.
Lines 9700	Onwards are the data loader and reader.

Variables

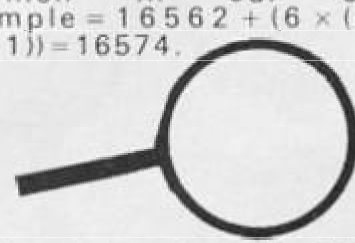
PPA + PPB	Players screen position.
CPA + CPB	Computers screen position.
TA + TB	Computers target screen position.
PR + TR	Computers present (or last) room and target room.
DP	Data pointer.
LR	Last room player was in.
D	The dice value.
E	Used as a counter.
N + V + L	Various loops.
G(3)	The present guess of the player or the computer.
A(3)	The answer.
X(3)	Used to set up the cards.
0(3,3)	The computers note pad.
C(2,3,3)	Holds the cards.
R$(7,8)	
W$(7,8)	
N$(7,8)	



Any mistakes in the data will cause the computer to move in the wrong directions and may even cause an error with report code 5. To check the data RUN 9800 and correct any mistakes.

For those interested in the layout and use of the data, it is stored in 8 blocks of 42 bits of data. The first block is for the computer to move from the start, the second block is for the computer to move from room 1 and so on up to the 8th block for the computer to move from room 7. These blocks are divided into 7 sections of 6 bits of data. The 7 sections are for the room the computer is going to e.g. block 0 section 3 is for the computer to move from the start to room 3. The sections are further split into 3 pairs, each pair consisting of line number and column number.

For an example of how the computer uses the data let us suppose the computer is in room 1 and wishes to go to room 3. First it sets the data pointer to the start of the data which is 16520, it then adds 42 × (the present room no:) which gives 16562 then adds 6 × ((target room no:) − 1) which in our example = 16562 + (6 × (3 − 1)) = 16574.



The computer then takes PEEK 16574 as the line number and PEEK (16574 + 1) as the column number. When making its move it moves in the direction of these coordinates until it reaches them. When it has reached a target it checks to see if it is in a room. If it is not in a room the data pointer is incremented by 2 and the next line and column numbers are got from the data. When the computer gets to a room it makes a guess and then sets the present room value and starts the whole process again after picking a new target room.

As a final note it is best to leave the data loader and reader part of the program intact in case an error is found at a later date. To save the program with Auto-start RUN 9600.

Happy Sleuthing.

Listing 1











```
   0 REM DATA HELD HERE-SEE TEXT

   1 RAND
   2 CLS
   3 SLOW
   9 REM SET UP
  10 LET PPA=10
  11 LET PPB=15
  15 LET CPA=10
  16 LET CPB=16
  17 LET TA=10
  18 LET TB=16
  19 LET PR=0
  20 LET TR=0
  21 LET DP=16520
  22 LET LR=0
  30 DIM N$(7,8)
  31 DIM R$(7,8)
  32 DIM W$(7,8)
  33 DIM A(3)
  34 DIM C(2,3,3)
  35 DIM G(3)
  36 DIM O(3,7)
  50 LET N$(1)="PAUL"
  51 LET N$(2)="JOHN"
  52 LET N$(3)="MARK"
  53 LET N$(4)="PAT"
  54 LET N$(5)="STEVE"
  55 LET N$(6)="SARA"
  56 LET N$(7)="TRACY"
  61 LET R$(1)="KITCHEN"
  62 LET R$(2)="CLOSET"
  63 LET R$(3)="BEDROOM"
  64 LET R$(4)="HALL"
  65 LET R$(5)="LOUNGE"
  66 LET R$(6)="STAIRS"
  67 LET R$(7)="BATHROOM"
  71 LET W$(1)="STICK"
  72 LET W$(2)="HAMMER"
  73 LET W$(3)="AXE"
  74 LET W$(4)="KNIFE"
  75 LET W$(5)="GUN"
  76 LET W$(6)="LEADPIPE"
  77 LET W$(7)="ROPE"
  79 PRINT TAB 13;"DOODUNNIT"
  80 PRINT ,,"YOU HAVE TO GUESS
WHICH ROOM THECRIME WAS COMMITTE
D IN, WHICH   WEAPON WAS USED AN
D WHO DID IT."
  81 PRINT "TO MAKE A GUESS YOU
MUST BE IN AROOM. THE ROOM YOU A
RE IN IS THE"
  82 PRINT "GUESS FOR THE ROOM.
ENTER YOUR  GUESS FOR THE WEAPON
 AND NAME OFTHE PERSON AS A SING
LE NUMBER."
  83 PRINT "YOU MAY NOT GO BACK
INTO A ROOM UNTIL YOU HAVE BEEN
INTO ANOTHER"
  85 PRINT ,,"USE THE CURSOR KEY
S TO MOVE      LEFT,RIGHT,UP AND
DOWN, AND KEYS1,2,3 AND 4 TO MOV
E DIAGONALLY."
  86 PRINT "YOU ARE THE * AND TH
E COMPUTER  IS THE C."
  87 GOSUB 9300
  88 CLS
  98 FAST
  99 REM DEAL
 100 LET E=1
 120 DIM X(7)
 130 LET B=0
 140 GOSUB 9060
 150 LET A(E)=D
```

```
160 LET X(D)=1
170 FOR N=1 TO 6
180 LET NN=INT (N/2+.5)
190 GOSUB 9060
200 IF X(D)<>0 THEN GOTO 180
210 LET C(B+1,E,NN)=D
220 LET X(D)=1
225 IF B=0 THEN LET O(E,D)=1
230 LET B=NOT B
240 NEXT N
250 LET E=E+1
260 IF E<4 THEN GOTO 110
270 SLOW
280 GOSUB 9200
285 GOSUB 9300
290 GOSUB 1000
999 GOSUB 2000
```



Listing 2 Main program

```
1000 REM PLAN
1010 CLS
1014 REM
1015 REM NUMBERS 1-7 SHOULD BE
        IN INVERSE PRINT
1016 REM
1020 PRINT AT 0,0;"████████████
████████████████ KITCHEN  █
CLOSET █  BEDROOM ██      █
   2   █          ██      █
       █       3  █"
1030 PRINT "█  1        █
█       ██    ██ ██████ ██
██ █    █"
1040 PRINT "█      █
  █     ██     █
  █████████████
  █ HALL █"
1050 PRINT "█
     4   █████████
  █      ██ BATH █
  ██████████"
1060 PRINT "█ ROOM █
  █      ██     █
  █      ██ 7
  █      █"
1070 PRINT "█      █
  █      ██     █      █████
████LOUNGE ██████████
         ██STAIRS      █
         ██   6        █
     5   ██            █
         █"
1080 PRINT "██████████████████
██████████"
1100 PRINT AT PPA,PPB;"*";AT CPA
,CPB;"C"
1110 RETURN
```

```
1999 STOP
2000 REM YOUR MOVEMENT
2010 GOSUB 9010
2020 PRINT AT 0,0;"YOUR MOVE. DI
CE=";D
2030 GOSUB 9030
2035 PRINT AT 0,0;"██████████████
██████████████████"
2040 FOR Y=1 TO D
2050 LET M$=INKEY$
2060 IF M$="" THEN GOTO 2050
2070 PRINT AT (PPA+(M$="6")+(M$=
"3")+(M$="4")-(M$="1")-(M$="2")-
(M$="7")),(PPB+(M$="2")+(M$="3")
+(M$="8")-(M$="1")-(M$="4")-(M$=
"5"));
2075 LET P=PEEK (PEEK 16398+256*
PEEK 16399)
2080 PRINT AT PPA,PPB;" "
2085 IF P<>0 THEN GOTO 3000
2090 LET PPA=PPA+(M$="6")+(M$="3
")+(M$="4")-(M$="1")-(M$="2")-(M
$="7")
2095 LET PPB=PPB+(M$="2")+(M$="3
")+(M$="8")-(M$="1")-(M$="4")-(M
$="5")
2100 GOSUB 1100
2110 NEXT Y
2120 GOSUB 1020
2200 GOTO 4000
2999 STOP
3000 REM IN ROOM
3005 IF P<64 THEN GOTO 2090
3010 IF P>156 THEN GOTO 3100
3020 GOSUB 1100
3025 SLOW
3030 GOTO 2050
3100 LET R=P-156
3105 IF R=LR THEN GOTO 3020
3110 LET LR=R
3115 LET P=0
3120 FAST
3125 CLS
3130 PRINT "YOU ARE ";"IN THEON
THE"(6*(R=6)+1 TO 6*(R=6)+6);" "
;R$(R)
3140 PRINT ,,"ENTER YOUR GUESS,
WEAPON NUMBER AND NAME NUMBER FR
OM :-"
3145 GOSUB 9110
3150 GOSUB 9200
3155 INPUT G$
3160 IF LEN G$<>2 OR CODE G$(1)<
29 OR CODE G$(2)<29 OR CODE G$(1
)>35 OR CODE G$(2)>35 THEN GOTO
3150
3155 INPUT G$
3160 IF LEN G$<>2 OR CODE G$(1)<
29 OR CODE G$(2)<29 OR CODE G$(1
```



```
)>35 OR CODE G$(2)>35 THEN GOTO
3150
3175 CLS
3180 LET G(1)=R
3190 LET G(2)=VAL G$(1)
3195 LET G(3)=VAL G$(2)
3200 FOR V=1 TO 3
3210 IF A(V)<>G(V) THEN GOTO 350
0
3220 NEXT V
3230 PRINT "YOU GOT IT RIGHT"
3250 GOTO 9400
3500 REM WRONG
3501 LET Q=0
3510 LET V=INT (RND*3)+1
3520 FOR X=1 TO 3
3530 IF C(1,V,X)=G(V) THEN GOTO
3600
3541 LET Q=Q+1
3542 IF Q=3 THEN GOTO 3560
3545 LET V=V+1
3549 NEXT X
3550 IF V=4 THEN LET V=1
3555 GOTO 3520
3560 PRINT "WRONG BUT I HAVE NON
E OF THEM"
3565 SLOW
3570 GOSUB 9300
3575 GOSUB 1000
3580 GOTO 2079
3600 PRINT "WRONG, I HAVE "
3601 IF V=1 THEN PRINT R$(G(V))
3602 IF V=2 THEN PRINT W$(G(V))
3603 IF V=3 THEN PRINT N$(G(V))
3605 SLOW
3610 GOSUB 9300
3611 GOSUB 1000
3615 GOTO 2079
3999 STOP
4000 REM COMPUTER MOVE
4001 GOSUB 9000
4003 PRINT AT 0,0;"MY MOVE. DICE
=";D
4004 GOSUB 9030
4005 PRINT AT 0,0;"██████████████
███"
4010 FOR Y=1 TO D
4020 LET MA=0+(CPA<TA)-(CPA>TA)
4030 LET MB=0+(CPB<TB)-(CPB>TB)
4035 LET CPA=CPA+MA
4036 LET CPB=CPB+MB
4040 IF CPA=TA AND CPB=TB THEN G
OTO 5000
4050 PRINT AT CPA-MA,CPB-MB;" "
4070 GOSUB 1100
4080 NEXT Y
4090 GOSUB 1020
4100 GOTO 2000
4499 REM DATA SORT
4500 LET PR=TR
4510 FOR J=1 TO 7
4520 LET TR=INT (RND*7)+1
4530 IF O(1,TR)<>0 THEN NEXT J
4540 IF TR=PR THEN LET TR=INT (R
ND*7)+1
4545 IF TR=PR THEN GOTO 4540
4550 IF PR<>0 THEN GOSUB 6000
4620 LET DP=16520+(PR*42)+((TR-1
)*6)
4630 LET TA=PEEK DP
4640 LET TB=PEEK (DP+1)
4645 PRINT AT 0,0;
4650 GOSUB 1020
4655 GOTO 4070
4999 STOP
5000 IF TR=0 THEN GOTO 4510
5010 PRINT AT CPA,CPB;
5015 LET P=PEEK (PEEK 16398+256*
PEEK 16399)
5016 IF P>128 THEN GOTO 4500
5017 IF P=23 THEN GOTO 5100
5020 LET DP=DP+2
5025 PRINT AT CPA-MA,CPB-MB;" "
5030 GOTO 4630
5100 LET CPA=CPA-MA
5110 LET CPB=CPB-MB
5120 GOTO 2000
5999 STOP
6000 REM COMPUTER GUESS
6010 CLS
6020 PRINT "MY GUESS :-"
6021 FAST
6025 LET E=2
6026 LET GD=INT (RND*7)+1
6030 FOR J=1 TO 7
6035 LET GD=GD+1
6036 IF GD>7 THEN LET GD=1
6040 LET G(E)=GD
6050 IF O(E,GD)<>0 THEN NEXT J
6060 LET E=E+1
6070 IF E<4 THEN GOTO 6026
6075 LET G(1)=PR
6080 PRINT
6090 PRINT G(1);" ";R$(G(1));" "
;G(2);" ";W$(G(2));" ";G(3);" ";
N$(G(3))
6095 GOTO 7000
6100 GOSUB 9200
6105 LET J=0
6110 PRINT ,,"IF YOU HAVE ANY OF
 MY GUESS      PLEASE ENTER ROOM
WEAPON OR NAME"
6115 PRINT "IF YOU HAVE NONE THE
N PRESS N/L"
6120 INPUT G$
6121 IF G$="" THEN GOTO 6500
6130 IF G$(1)="R" THEN LET J=1
6131 IF G$(1)="W" THEN LET J=2
```

```
6132 IF G$(1)="N" THEN LET J=3
6140 IF J=0 THEN GOTO 6120
6170 LET O(J,G(J))=1
6175 PRINT AT 15,0;"THANK YOU
          "
6176 SLOW
6180 GOSUB 9300
6190 RETURN
6500 LET E=1
6510 FOR L=1 TO 3
6520 IF C(2,E,L)=G(E) THEN GOTO
6600
6530 NEXT L
6540 LET E=E+1
6550 IF E<4 THEN GOTO 6510
6560 GOTO 6175
6600 PRINT AT 15,0;"CHEAT YOU H
AVE GOT ONE OF THEM "
6610 GOTO 6120
6999 STOP
7000 FOR J=1 TO 3
7010 IF G(J)<>A(J) THEN GOTO 610
0
7020 NEXT J
7030 PRINT "I GUESS RIGHT"
7040 GOTO 9400
8999 STOP
9000 REM SUBS
9010 LET D=INT (RND*6)+1
9020 RETURN
9030 FOR L=1 TO 10
9040 NEXT L
9050 RETURN
9060 LET D=INT (RND*7)+1
9070 RETURN
9100 CLS
9110 PRINT
9115 PRINT TAB 3;"ROOM";TAB 14;"
WEAPON";TAB 24;"NAME"
9120 FOR N=1 TO 7
9130 PRINT N;" ";R$(N);" ";N;" "
;W$(N);" ";N;" ";N$(N)
9140 NEXT N
9150 RETURN
9200 PRINT ,,"YOU HAVE :-"
9210 PRINT TAB 3;"ROOM";TAB 14;"
WEAPON";TAB 24;"NAME"
9220 PRINT
9230 FOR N=1 TO 3
9240 PRINT C(2,1,N);" ";R$(C(2,1
,N));" ";C(2,2,N);" ";W$(C(2,2,N
));" ";C(2,3,N);" ";N$(C(2,3,N))

9250 NEXT N
9260 RETURN
9300 PRINT ,,"PRESS ANY KEY TO C
ONTINUE"
9310 IF INKEY$<>"" THEN GOTO 931
0
9315 IF INKEY$="" THEN GOTO 9315

9320 RETURN
9400 PRINT ,,"THE MURDER WAS COM
MITTED BY",N$(G(3))
9410 PRINT "IN ON "(2*(1 AND G(1
)=6)+1 TO 3+(3*(G(1)=6)));"THE "
;R$(G(1))
9420 PRINT "WITH A ";W$(G(2))
9500 PRINT ,,"DO YOU WANT ANOTHE
R GAME (Y/N)?"
9510 INPUT A$
9520 IF A$="N" THEN PRINT "OK BY
 ME."
9525 IF A$="N" THEN STOP
9530 IF A$="Y" THEN RUN
9599 REM SAVE
9600 SAVE "OODUNNIT"
9610 RUN
9700 REM LOADER
9701 FAST
9705 LET A$=""
9710 PRINT "ENTER STARTING ADDRE
SS"
9715 INPUT A
9720 SCROLL
9725 PRINT A;":";
9730 FOR N=1 TO 6
9735 IF A$="" THEN INPUT A$
9740 POKE A,(16*CODE A$+CODE A$(
2)-476)
9745 PRINT " ";A$( TO 2);
9750 LET A=A+1
9755 LET A$=A$(3 TO )
9760 NEXT N
9765 PRINT
9770 GOTO 9720
9800 REM READER
9805 PRINT "ENTER STARTING ADDRE
SS"
9810 INPUT A
9815 PRINT A;":";
9820 FOR N=1 TO 6
9825 LET P=PEEK A
9830 PRINT " ";CHR$ (28+INT (P/1
6));CHR$ (28+P-(16*INT (P/16)));

9835 LET A=A+1
9840 NEXT N
9845 IF PEEK 16442=3 THEN GOTO 9
860
9850 PRINT
9855 GOTO 9815
9860 PRINT AT 0,0;"      PRESS AN
Y KEY TO SCROLL     "
9865 IF INKEY$="" THEN GOTO 9865

9870 SCROLL
9875 GOTO 9815
```

Glossyairy

or the Plain Man's Guide to Computer Terminology according to Jim Lock of Surrey.

Access	— sometimes used to buy a computer
Adaptor	— connector enabling other people's equipment to be attached to (but not work with) your computer
Addictive	— word frequency used by software reviewers to describe games their children wouldn't let them play
Adventure	— fantasy game requiring fairly logical handling of very unusual objects
Amplifier	— device which plugs into your computer and allows your neighbours to hear your children playing "3D PacVaders"
Arcade game	— colourful and noisy reaction tester
Array	— what you shout when your program works first time
Artificial intelligence	— what you think your computer has and it thinks you have
Autostart	— the aliens fire at you before you've put your fingers on the control keys
BASIC	— a standard language most suitable for newcomers to computing as it uses simple words and instructions such as ABS, OPEN #, LPRINT, USR, DEF FN and CHR$
BBC	— the little noise underneath a Sinclair computer
Bug	— a small creature, about the size of a gremlin, which crawls into the computer through a hole in the back and plays havoc with your programs
Command	— an order you give the computer which often gets the reply "Nonesense in BASIC"
Compatible	— it won't work with your ocmputer
Comprehensive	— contains absolutely everything you need to know, except for the one thing you want to find out
Computer tape	— used to attach a Sinclair RAMpack to a ZX81
Console	— flashy piece of furniture which neatly stores your computer, power supply, cables and accessories but takes up three times the space they did
Cursor key	— what you often do when you accidentally move your laser base in the wrong direction
Digital	— only works when poked hard with a finger
DIM	— how you feel when you see programs written by your seven year old daughter
Flexible	— unworkable, as in "flexible operating system"
Flying Simulation	— immensely complicated game requiring the use of at least fourteen fingers at once
FOR . . . NEXT	— short extract from conjuror's patter, "For my Next trick"
High resolution graphics	technique used by a computer to draw very wobbly diagonal lines on a T.V. screen
IF . . . THEN	conditional statement, as in "IF you try to copy this program THEN your computer will explode
Interference	— when your children tell you how to play "3D PacVaders"
Joystick	— gadget which will probably not work with your latest game
Kilobyte	— enormous hamburger
LEN and VAL	— the married couple who live around the corner
Listing	— what your computer is doing to one side, having lost one of it's rubber feet
Logic	what makes computers and male humans work (it's counterpart in female humans is intuition)
Low resolution graphics	— technique used by a computer to make almost recognisable pictures out of building blocks
Memory expansion	— something you discover you need a few days after buying the cheaper computer
Menu	— a list of options from which you invariably choose wrongly
Microdrive	— car-racing arcade game for your computer
Micronet	— used to catch bugs
Mind game	— intellectually demanding game requiring very unusual handling of fairly logical objects
Modem	— how assistants at Harrods address your girlfriend
Motherboard	— Mum's reaction to news of offspring's latest hi-score on "3D PacVaders"

NOT — used to tie two STRINGs together

Prompt — when you tell your children how to play "3D PacVaders"

RAMpack — device you plug in to the back of your computer so that it can uplug itself whilst you are typing in the last line of a very long program

Random Access Memory — what you need to find the interesting article you cut out of a magazine three months ago and put somewhere safe

REM — put at the beginning of a line to REMind you to remember it and the computer to ignore it

Royalties — The Queen and her family

Screwdriver — instrument for invalidating guarantees

Sinclair Printer — ingenious mechanical device for producing smudges on a roll of expensive paper and then shredding the lot before you get a chance to see what it has done

Sound chip — good golf shot

Speech synthesiser — box of extremely sophisticated electronic components which makes your computer sound like a Dalek

STRING used to attach a Sinclair RAMpack to a ZX81

Syntax error — "you are naughty, you forgot to declare your software royalties to the Inland Revenue"

Tool kit — large bag in which you keep your screwdriver

Uncopiable cassette — blank cassette

User defined graphic — only the guy who wrote the program knows what this funny shaped blob is supposed to be

Variable something which is fixed at the start of a program, changed during the program and reset to it's original value at the end of the program

Word Processing — operation which allows two computers to become pen pals



Netherwallop Golf Course

We have printed golf programs in the past, but I think you'll agree this one from Norman Brooks of Kent is superb.

Netherwallop Golf Course has 9 holes and is par 36.

You have 10 clubs in your bag ranging from a No. 1 Wood to a No. 9 Sand Wedge with your trusty Putter being club No. 0.

The game uses the score card as a menu screen and by pressing "N" you will move on move on through each hole in turn. For each hole, when displayed, you will see the hole no., the distance and par in a box at the top left of the screen. take care to note the distance as all holes look the same size on the screen but the flight of the ball is scaled to match the distance from tee to green. Choosng a No. 1 Wood on a 150 yard hole will throw you Out of Bounds easily!

You are first asked which club you wish to use (enter 1 to 9 or 0) and, dependent on your choice, the computer will recognise the maximum distance the ball can travel when hit with that club.

Secondly you are asked what swing you wish to employ (enter Nos. 1 to 4) Choosing 4 will identify a full blooded swing and the ball will travel the maximum distance. No. 3 selects a ¾ swing and the ball will travel ¾ of the max. distance, and so on down to 1.

Finally you are asked what direction you wish to hit the ball and you must enter an angle in degrees, clockwise from North. (There is a compass shown on the screen). Unfortunately the wind may be blowing and this can offset your chosen angle by up to + or − 15°.

As with all golf courses there are hazards and these are as follows:

- **"T" Trees** — On impact your ball stops dead, your next club is chosen for you and you only have choice of swing and angle.
- **"=" Water** — You only encounter this hazard if your ball ends up in thedrink at the end of its trajectory. You have to drop a new ball (penalty 1 stroke) and this will happen at the nearest dry spot to the West.



- **"✱" Rough** — You only encounter rough if your ball ends up in overgrowth at the end of its trajectory. Your next club is chosen for you and you only have choice of swing and angle.
- **"■" Bunker** — On impact your ball stops dead. You can only use your Sand Wedge to get out and your swing is chosen for you. You can choose your angle of exit.
- **Out Of Bounds** — Going into the black line surrounding the course puts you out of bounds. You must drop a new ball (penalty 1 stroke) where the old one was lost to continue.

Once you have reached the green, the screen changes to show the green only. Your ball is shown as an "0" and the hole is an "inverse 0". You only use your putter now but you can choose a putting strength from 1 to 20 and a putting angle as before. You will not be troubled by the wind. If you, not only get on the green, but land on the "+" character, then automatically when the green screen appears, you will be placed closer to the hole — or if you are really lucky you may "hole out" and not need to putt!

After a SUNK PUTT you will automatically return to the score card and will see how many strokes you took on that hole.

My course 'best ever' is a "one under par" 35 — beat that!

Program notes.

LOAD the program and then RUN to get started.

I have needed to employ 'Shoe-horn' methods to get the program into 16K involving (a) removing all REM statements and (b) converting numbers to VAL 'numbers' to conserve memory. The missing REM lines would read:

Line No.	Statement
100	Main Program.
2000	Bunker Hazard Routine.
2500	Rough Hazard Routine.
3000	Water Hazard Routine.
3500	Tree Hazard Routine.
4000	Out of Bounds Routine.
4500	Sunk Putt Routine.
5000	Green Routine.
6000	Clear Message Area.
6100	Print Course.
6200	Print Green.
6500-8740	Set Up each of 9 Holes as required.
8750	Draw Course Boundaries.
9000	Score Card.
9500	Introduction.
9750	Set Up Arrays.

Line No.	Statement
100	Main Program.
2000	Bunker Hazard Routine.
2500	Rough Hazard Routine.
3000	Water Hazard Routine.
3500	Tree Hazard Routine.
4000	Out of Bounds Routine.
4500	Sunk Putt Routine.
5000	Green Routine.
6000	Clear Message Area.
6100	Print Course.
6200	Print Green.
6500-8740	Set Up each of 9 Holes as required.
8750	Draw Course Boundaries.
9000	Score Card.
9500	Introduction.
9750	Set Up Arrays.

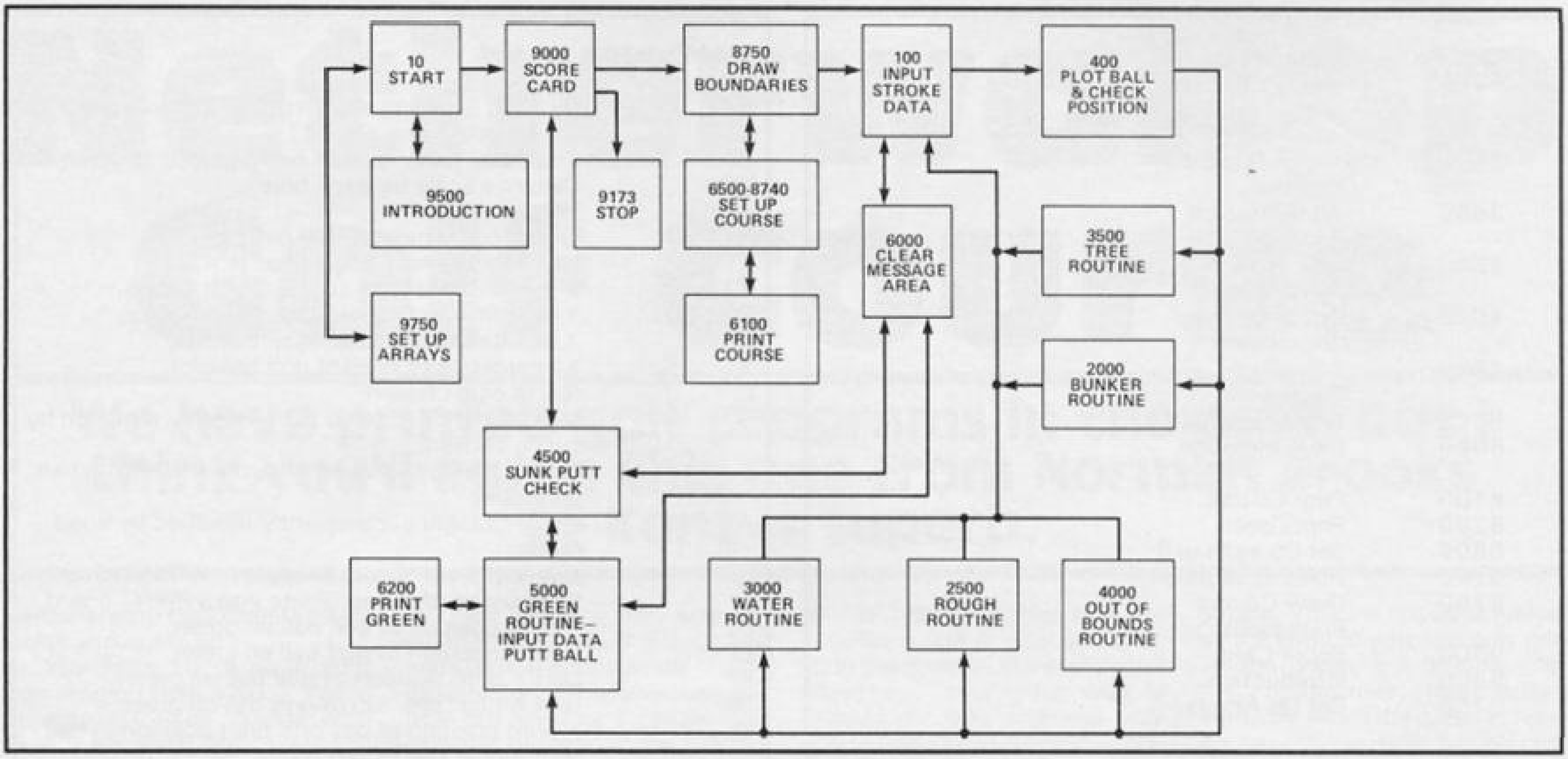


Variables used

S(9)	Numeric String to store strokes per hole.
N	No. of hole being played.
D$(15,25)	Character Array of hole being played or green.
Z(9)	Distance scale for each hole.
T	Total score.
X1	Last X coordinate of golf ball plot.
Y1	Last Y coordinate of golf ball plot.
X2	X coordinate of ball at end of last stroke.
Y2	Y coordinate of ball at end of last stroke.
X	X coordinate of current golf ball plot.
Y	Y coordinate of current golf ball plot.
C	No. of club chosen.
D	Max. distance for club subsequently modified by swing chosen.
P	Swing strength chosen or putter strength when on green.
A	Angle of flight subsequently modified by wind factor.
L	Conversion of Y coordinate into a PRINT coord.
K	Conversion of X coordinate into a PRINT coord.
B1	Y print position of golf ball on green.
B2	X print position of golf ball on green.
B3	Last Y print position of golf ball on green.
B4	Last X print position of golf ball on green.
B6	X print position of ball onY print position of ball on green at end of last putt.
01	Random Y print position of hole on green.
02	Random X print position of hole on green.

```
  40 GOSUB VAL "9500"
  50 GOTO VAL "9000"
 105 GOSUB VAL "6000"
 110 PRINT AT 2,10;"WHICH CLUB?
(0 TO 9)   "
 120 INPUT C
 125 IF C>9 THEN GOTO 120
 130 IF NOT C THEN LET D=20
 140 IF C>0 AND C<=4 THEN LET D=
270-C*30
 150 IF C>4 THEN LET D=220-C*20
 155 PRINT AT 2,10;"
           ";AT 2,10;
 160 IF NOT C THEN PRINT "PUTTER
";
 170 IF C>0 AND C<=2 THEN PRINT
"NO.";C;" WOOD";
 180 IF C>2 AND C<=8 THEN PRINT
"NO.";C;" IRON";
 190 IF C>8 THEN PRINT "SAND WED
GE";
 195 PRINT " SELECTED"
 210 PRINT AT 3,10;"WHAT SWING?
(1 TO 4)   "
 220 INPUT P
 225 IF P<1 OR P>4 THEN GOTO 220

 255 IF C THEN PRINT AT 3,10;"CH
OICE IS A "+("CHIP SHOT " AND P=
1)+("1/2 " AND P=2)+("3/4 " AND
P=3)+("FULL" AND P=4)+(" SWING"
AND P>1)
 280 IF NOT C THEN PRINT AT 3,10
;"FORCE ";P;" PUTT SELECTED "
```




```
 290 LET D=D*P/4
 310 PRINT AT 4,10;"DEG.CLOCKWIS
E FROM N? "
 320 INPUT A
 322 IF A<0 OR A>360 THEN GOTO 3
20
 325 PRINT AT 4,10;(" " AND A<10
)+(" " AND A<100);A;" DEG.     "
 328 LET R=15-INT (1+30*RND)
 330 IF R THEN PRINT AT 4,19;"CR
OSSWIND"+("+" AND R>0);R;" "
 331 IF NOT R THEN PRINT AT 4,19
;"NO CROSSWIND "
 335 LET A=A+R
 345 LET A=A+360*((-(A>360))+(A<
0))
 350 LET A=PI*A/180
 410 LET S(N)=S(N)+1
 460 FOR R=0 TO D/Z(N) STEP 2
 470 LET X=X2+R*SIN A
 480 LET Y=Y2+R*COS A
 485 IF X<-0.5 OR X>=49.5 OR Y<-
0.5 OR Y>=29.5 THEN GOTO 4000
 500 UNPLOT X1+8,Y1+2
 520 LET L=15-INT ((Y1+0.5)/2)
 530 LET K=1+INT ((X1+0.5)/2)
 540 PRINT AT 5+L,3+K;D$(L,K)
 545 PLOT X+8,Y+2
 550 IF R<>0 AND D$((15-INT ((Y+
0.5)/2)),(1+INT ((X+0.5)/2)))="T
" THEN GOTO 3500
 560 IF R<>0 AND D$((15-INT ((Y+
0.5)/2)),(1+INT ((X+0.5)/2)))="▒
" THEN GOTO 2000
 610 LET X1=X
 620 LET Y1=Y
 730 NEXT R
 740 LET X2=X
 750 LET Y2=Y
 754 LET L=15-INT ((Y+0.5)/2)
 755 LET K=1+INT ((X+0.5)/2)
 756 IF D$(L,K)="▒" THEN LET B1=
1+INT (RND*15)
 757 IF D$(L,K)="▒" THEN LET B2=
1+INT (RND*25)
 758 IF D$(L,K)="+" THEN LET B1=
11+INT (RND*5)
 759 IF D$(L,K)="+" THEN LET B2=
16+INT (RND*5)
 760 IF R<>0 AND D$(L,K)="▒" THE
N GOTO 5000
 765 IF R<>0 AND D$(L,K)="+" THE
N GOTO 5000
 770 IF R<>0 AND D$(L,K)="=" THE
N GOTO 3000
 780 IF R<>0 AND D$(L,K)="*" THE
N GOTO 2500
 990 GOTO VAL "100"
2010 GOSUB VAL "6000"
2020 LET X2=X
2025 LET Y2=Y
2030 LET X1=X
2035 LET Y1=Y
2040 LET D=40
2050 PRINT AT 2,10;"BUNKER-SAND
WEDGE ONLY"
2060 LET P=1+INT (RND*4)
2070 IF P=2 THEN PRINT AT 3,10;"
YOU CAN ONLY CHIP OUT."
2080 IF P=3 THEN PRINT AT 3,10;"
YOU CAN ONLY SLICE OUT"
2090 IF P=1 THEN PRINT AT 3,10;"
OH DEAR...DREADFUL LIE"
2100 IF P=4 THEN PRINT AT 3,10;"
YOU CAN ONLY BLAST OUT"
```

```
2490 GOTO VAL "290"
2510 GOSUB VAL "6000"
2600 LET C=4+INT (RND*6)
2700 PRINT AT 2,10;"ROUGH-YOUR L
IE IS "+("GOOD" AND C<6)+("FAIR"
 AND (C>5 AND C<8))+("POOR" AND
C>7)
2980 PAUSE 150
2990 GOTO VAL "140"
3010 LET S(N)=S(N)+1
3020 GOSUB VAL "6000"
3030 PRINT AT 2,10;"YOU ARE IN
THE WATER ";TAB 10;"PENALTY = 1
STROKE....";TAB 10;"PUSH N TO DR
OP NEWBALL"
3040 IF INKEY$<>"N" THEN GOTO 30
40
3050 UNPLOT X+8,Y+2
3130 FOR R=1 TO 20
3140 LET X=X-2
3160 LET L=15-INT ((Y+0.5)/2)
3170 LET K=1+INT ((X+0.5)/2)
3180 IF D$(L,K)=" " THEN GOTO 32
00
3190 NEXT R
3200 LET X2=X
3210 LET Y2=Y
3220 PLOT X+8,Y+2
3490 GOTO VAL "100"
3520 GOSUB VAL "6000"
3580 LET X2=X
3590 LET Y2=Y
3600 LET C=5+2*INT (RND*3)
3950 PRINT AT 2,10;"TREES-YOUR L
IE IS     "
3954 PRINT AT 2,28;("GOOD" AND C
=5)+("FAIR" AND C=7)+("POOR" AND
 C=9)
3960 PAUSE 150
3990 GOTO VAL "150"
4010 LET S(N)=S(N)+1
4020 GOSUB VAL "6000"
4050 UNPLOT X1+8,Y1+2
4060 LET X=X1
4070 LET Y=Y1
4080 LET X2=X
4090 LET Y2=Y
4450 PRINT AT 2,10;"YOU ARE OUT
OF BOUNDS "
4460 PRINT AT 3,10;"PENALTY = 1
STROKE...."
4470 PRINT AT 4,10;"PUSH N TO DR
OP NEWBALL"
4480 IF INKEY$<>"N" THEN GOTO 40
80
4485 PLOT X+8,Y+2
4487 PRINT AT 4,10;"
          "
4490 GOTO VAL "100"
4520 IF D$(B1,B2)<>"▣" THEN RETU
RN
4530 GOSUB VAL "6000"
4540 PRINT AT 2,10;"▛▌▐▛"+CHR$
 4+" ▌▛"+CHR$ 135+CHR$ 1+"   ▛▌▌
"+CHR$ 2+"▛"+CHR$ 2+"▛"
4550 PRINT TAB 10;"▙▌▌▌"+CHR$
2+CHR$ 4+"▌▙   ▛▌▌▌ ▌"
4560 PRINT TAB 10;"▟▙▌ "+CHR$
 2+"▟▌▚ ▌ ▙▌▌"
4570 FOR R=1 TO 25
4580 PRINT AT 5+B1,3+B2;"▣";AT 5
+B1,3+B2;"O"
4600 NEXT R
4990 GOTO VAL "9000"
5040 GOSUB VAL "6200"
5045 IF D$(L,K)="▣" THEN GOTO 45
30
5047 PRINT AT 10+O1,18+O2;D$(5+O
1,15+O2)
5082 PRINT AT 5+B1,3+B2;"O"
5084 LET B5=B1
5085 LET B3=B1
5086 LET B4=B2
5087 LET B6=B2
5090 GOSUB VAL "4500"
5095 LET D=20
5097 GOSUB VAL "6000"
5098 PRINT AT 2,10;"GREEN-PUTTER
 SELECTED "
5100 PRINT AT 3,10;"STRENGTH? (1
 TO 20)    "
5110 INPUT P
5120 LET D=D*P/20
5125 IF P<1 OR P>20 THEN GOTO 51
10
5130 PRINT AT 3,10;"FORCE "+(" "
 AND P<10);P;" PUTT SELECTED"
5150 PRINT TAB 10;"DEG.CLOCKWISE
 FROM ▣? "
5155 INPUT A
5160 IF A<0 OR A>360 THEN GOTO 5
155
5170 PRINT AT 4,10+(A<100)+(A<10
);A;" DEG. CHOSEN        "
5200 IF A<0 THEN LET A=A+360
5210 IF A>360 THEN LET A=A-360
5220 LET A=PI*A/180
5230 LET S(N)=S(N)+1
5250 FOR R=1 TO D
5260 LET B1=B5-INT (R*COS A)
5270 LET B2=B6+INT (R*SIN A)
5272 IF B1<1 THEN LET B1=1
5274 IF B1>15 THEN LET B1=15
5276 IF B2<1 THEN LET B2=1
5278 IF B2>25 THEN LET B2=25
5280 PRINT AT 5+B3,3+B4;"▒"
5290 PRINT AT 5+B1,3+B2;"O"
5300 GOSUB VAL "4500"
```

```
5302 LET B3=B1
5304 LET B4=B2
5310 NEXT R
5320 LET B5=B1
5330 LET B6=B2
5490 GOTO VAL "5095"
6010 FOR R=2 TO 4
6020 PRINT AT R,10;"
     "
6030 NEXT R
6040 RETURN
6110 FOR R=1 TO 15
6120 PRINT AT 5+R,4;D$(R)
6130 NEXT R
6140 RETURN
6210 FOR R=1 TO 15
6220 LET D$(R)="▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒▒
▒▒▒▒▒▒▒▒"
6230 PRINT AT 5+R,4;D$(R)
6240 NEXT R
6250 LET O1=1+INT (RND*5)
6260 LET O2=1+INT (RND*5)
6270 LET D$(5+O1,15+O2)="▣"
6290 RETURN
6510 GOSUB VAL "8500"
6520 FOR R=1 TO 3
6530 LET D$(1+R,21 TO 24)="    "
6540 LET D$(6+R,17 TO 20)="▒▒▒▒"
6550 NEXT R
6560 LET D$(8,19)="+"
6570 PRINT AT 1,6;9;AT 2,6;4;AT
3,5;450
6580 PRINT AT 0,13;" THE HEATH 
"
6590 GOSUB VAL "6100"
6600 LET Z(N)=VAL "11.271153"
6620 RETURN
6760 GOSUB VAL "7500"
6770 FOR R=1 TO 3
6780 LET D$(10+R,20 TO 23)="
"
6790 LET D$(2+R,19 TO 22)="▒▒▒▒"
6800 NEXT R
6810 LET D$(4,21)="+"
6820 PRINT AT 1,6;8;AT 2,6;4;AT
3,5;400
6830 PRINT AT 0,13;" BLACK SHAMP
█ "
6840 GOSUB VAL "6100"
6850 LET Z(N)=VAL "8.2026478"
6860 RETURN
7010 GOSUB VAL "7250"
7020 FOR R=1 TO 3
7025 LET D$(1+R,21 TO 24)="    "
7030 LET D$(6+R,19 TO 22)="▒▒▒▒"
7040 NEXT R
7050 LET D$(8,21)="+"
7100 PRINT AT 1,6;7;AT 2,6;3;AT
3,5;200
7110 PRINT AT 0,13;" THE SANDPIT
█ "
7120 GOSUB VAL "6100"
7130 LET Z(N)=VAL "4.5810871"
7140 RETURN
7260 LET D$(1)="==    ***********
***    "
7265 LET D$(2)=" ===   **********
    ▒▒▒▒ "
7270 LET D$(3)="    ==  ****
    ▒▒+▒ "
7275 LET D$(4)="     ==  **    T
    ▒▒▒▒ "
7280 LET D$(5)="     ==
 T      "
7285 LET D$(6)="T    ==    T
   ▒ *"
7290 LET D$(7)="TT     =
     *"
7295 LET D$(8)="TTT     ==
T      **"
7300 LET D$(9)="TTT      =====
     **"
7305 LET D$(10)="TT       =====
   ▒  **"
7310 LET D$(11)="T          ======
  ▒▒ ***"
7315 LET D$(12)="              =
=    *****"
7320 LET D$(13)="
===*******"
7325 LET D$(14)="              *
**==******"
7330 LET D$(15)="          ********
***=*****"
7340 IF N=7 THEN RETURN
7400 PRINT AT 1,6;6;AT 2,6;5;AT
3,5;530
7410 PRINT AT 0,13;" THE SHEEPWA
SH  "
7420 GOSUB VAL "6100"
7440 LET Z(N)=VAL "10.29563"
7450 RETURN
7510 LET D$(1)="TTTTT********=**
*****=***"
7515 LET D$(2)="TTTT********* ==
=====****"
7520 LET D$(3)="TTT****      **
**    ***"
7525 LET D$(4)="TTT**      ▒
     ***"
7530 LET D$(5)="TTT*
      **"
7535 LET D$(6)="TT*
   ▒ **"
```

```
7540 LET D$(7)="TT*
  ▒▒   *"
7545 LET D$(8)="TT
       *"
7550 LET D$(9)="T
      T"
7555 LET D$(10)="T          TTTT
      T"
7560 LET D$(11)="         TTTTTTT
  ▒▒▒▒ T"
7565 LET D$(12)="       TTTTTTTTT
 ▒▒ ▒▒+▒ T"
7570 LET D$(13)="     TTTTTTTTTTT
  ▒▒▒▒ T"
7575 LET D$(14)="    TTTTTTTTTTTT
T       T"
7580 LET D$(15)=" TTTTTTTTTTTTT
TTTTTTTTTT"
7590 IF N=8 THEN RETURN
7600 PRINT AT 1,6;5;AT 2,6;4;AT
3,5;380
7610 PRINT AT 0,13;" CUCKOO WOOD
S "
7620 GOSUB VAL "6100"
7640 LET Z(N)=VAL "8.7223898"
7650 RETURN
7760 LET D$(1)="
    "
7765 LET D$(2)="       T T
     ▒ "
7770 LET D$(3)="     T    T     T
T  ▒▒▒▒  "
7775 LET D$(4)="              TT▬
"+CHR$ 4+"T ▒▒+▒  "
7780 LET D$(5)="     T    T     T▛▙
▙TT▒▒▒▒   "
7785 LET D$(6)="      T T      T"+
CHR$ 135+"█"+CHR$ 4+"T     ▒ "
7790 LET D$(7)="                TTT
TTT      *"
7795 LET D$(8)="            T
      **"
7800 LET D$(9)="           T T
      **"
7805 LET D$(10)="***        T
      ***"
7810 LET D$(11)="****
     ****"
7815 LET D$(12)="*****
      ****"
7820 LET D$(13)="    ****
     *****"
7825 LET D$(14)="     ***
     *****"
7830 LET D$(15)="    ***
    ******"
7900 PRINT AT 1,6;4;AT 2,6;4;AT
3,5;440
7910 PRINT AT 0,13;" EAGLE ROCK 
 "
7920 GOSUB VAL "6100"
7940 LET Z(N)=VAL "9.0229126"
7950 RETURN
8010 LET D$(1)="************
  TTTT"
8015 LET D$(2)="*****           ===
====  TT"
8020 LET D$(3)="***            =====
=====  TT"
8025 LET D$(4)="**              ===
   ==  T"
8030 LET D$(5)="**             ===  ▒
▒▒▒  == T"
8035 LET D$(6)="**  ==     ==     ▒
▒+▒ ==  "
8040 LET D$(7)="*  ===  ==        ▒
▒▒▒  ===  "
8045 LET D$(8)="*  =======  ▒
    ===  "
8050 LET D$(9)="* ========
    ===   "
8055 LET D$(10)="* ====  ==
    ===    "
8060 LET D$(11)="*  ==     ==    ▒
  ===    T"
8065 LET D$(12)="               ===
  ===      T"
8070 LET D$(13)="               ====
====     TT"
8075 LET D$(14)="                 ==
==      TTT"
8080 LET D$(15)="
   TTTTTTT"
8090 PRINT AT 1,6;3;AT 2,6;3;AT
3,5;150
8100 PRINT AT 0,13;" BAMBOO ISLA
ND "
8200 GOSUB VAL "6100"
8210 LET Z(N)=VAL "3.7665147"
8240 RETURN
8260 LET D$(1)="   * * * * *
         "
8265 LET D$(2)="   * * * * *
T T T      "
8270 LET D$(3)="   * * * * *    T
       T  "
8275 LET D$(4)="   * * * * * T
  ▒        "
8280 LET D$(5)="   * * * * *    ▒
   ▒  T  "
8285 LET D$(6)="   * * * * *
         "
8290 LET D$(7)="   * * * * *
  ▒▒▒▒ T  "
8295 LET D$(8)="   * * * * *
  ▒▒+▒     "
8300 LET D$(9)="   * * * * *
  ▒▒▒▒ T  "
```

```
8305 LET D$(10)="  * * * * *
     "
8310 LET D$(11)="  * * * * *
  ▒    T "
8315 LET D$(12)="  * * * * * T
     "
8320 LET D$(13)="  * * * * *   T
     T  "
8325 LET D$(14)="  * * * * *
 T T T  "
8330 LET D$(15)=" * * * * *
     "
8400 PRINT AT 1,6;2;AT 2,6;4;AT
3,5;370
8410 PRINT AT 0,13;" THE HEDGES
"
8420 GOSUB VAL "6100"
8440 LET Z(N)=VAL "8.4750111"
8450 RETURN
8510 LET D$(1)="******* TTTTTTTT
T  "
8515 LET D$(2)="****      TTTTTT
    ▒▒▒▒ "
8520 LET D$(3)="**
    ▒▒+▒ "
8525 LET D$(4)="*
▒▒▒   ▒▒▒▒ "
8530 LET D$(5)="*      T
     "
8535 LET D$(6)="     TTT
   ▒▒  "
8540 LET D$(7)="       T       TT"
8545 LET D$(8)="              TTTT
        *"
8547 LET D$(9)="              TT
      **"
8550 LET D$(10)="
     ****"
8555 LET D$(11)="      ===
 *********"
8560 LET D$(12)="     ====     ***
**********"
8565 LET D$(13)="     ===    ******
**********"
8570 LET D$(14)="         *********
**********"
8575 LET D$(15)="      **********
**********"
8577 IF N=9 THEN RETURN
8580 PRINT AT 1,6;1;AT 2,6;5;AT
3,5;500
8590 PRINT AT 0,13;" RABBIT WARR
EN "
8600 GOSUB VAL "6100"
8630 LET Z(N)=VAL "9.7128586"
8740 RETURN
8755 CLS
8760 PRINT "▛▀▀▀▀▀▀▀▀▀▜"
8770 PRINT "▌HOLE   . ▐"
8780 PRINT "▌PAR      ▐"
8790 PRINT "▌YDS      ▐"
8800 PRINT "▙▄▄▄▄▄▄▄▄▄▟"
8810 PRINT TAB 3;CHR$ 135+"▀▀▀▀
▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀"+CHR$ 4
8820 FOR R=1 TO 15
8830 PRINT TAB 3;"▌";TAB 29;"▐"
8840 NEXT R
8850 PRINT TAB 3;CHR$ 2+"▄▄▄▄▄
▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄"+CHR$ 1
8860 PRINT AT 20,0;"TEE"
8870 PRINT AT 11,1;"N";AT 12,0;"
W+E";AT 13,1;"S"
8980 GOSUB VAL "(8750-250*N)"
8990 GOTO VAL "100"
9010 CLS
9020 PRINT AT 0,0;" NETHERHALLOP
 GOLF   SCORE CARD "
9040 PRINT TAB 21;"YDS PAR OUT"
9050 PRINT " 1.  RABBIT WARREN
 500  5"
9052 PRINT
9055 PRINT " 2.  THE HEDGES
 370  4"
9057 PRINT
9060 PRINT " 3.  BAMBOO ISLAND
 150  3"
9062 PRINT
9065 PRINT " 4.  EAGLE ROCK
 440  4"
9067 PRINT
9070 PRINT " 5.  CUCKOO WOODS
 380  4"
9072 PRINT
9075 PRINT " 6.  THE SHEEPWASH
 530  5"
9077 PRINT
9080 PRINT " 7.  THE SANDPIT
 200  3"
9082 PRINT
9084 PRINT " 8.  BLACK SWAMP
 400  4"
9086 PRINT
9088 PRINT " 9.  THE HEATH
 450  4"
9095 PRINT TAB 25;"-- ---";TAB 1
8;"TOTALS 36";TAB 25;"-- ---"
9105 FOR R=1 TO 9
9110 PRINT AT 2*R,29+(S(R)<10);S
(R)
9120 NEXT R
9125 LET T=0
9130 FOR R=1 TO 9
9140 LET T=T+S(R)
9150 NEXT R
9165 PRINT AT 20,28+(T<100)+(T<1
0);T
```

```
9170 LET N=N+1
9172 IF N=10 THEN PRINT AT 21,0;
"GAME COMPLETE.              "
9173 IF N=10 THEN STOP
9175 PRINT AT 21,0;"PUSH ""N"" F
OR NEXT HOLE."
9180 IF INKEY$<>"N" THEN GOTO VA
L "9180"
9190 LET X1=0
9195 LET Y1=0
9200 LET X2=1
9205 LET Y2=1
9490 GOTO VAL "8750"
9510 PRINT AT 0,0;"     NETHERHAL
LOP GOLF COURSE    "
9512 PRINT
9515 PRINT TAB 10;"BY  N.BROOKS"
9520 PRINT
9530 PRINT "WELCOME....YOU HAVE
10 CLUBS...."
9540 PRINT
9550 PRINT TAB 1;"CODE";TAB 9;"C
LUB";TAB 17;"CODE";TAB 25;"CLUB"

9570 PRINT TAB 2;1;TAB 6;"NO.1 W
OOD";TAB 18;6;TAB 22;"NO.6 IRON"
9580 PRINT TAB 2;2;TAB 6;"NO.2 W
OOD";TAB 18;7;TAB 22;"NO.7 IRON"

9590 PRINT TAB 2;3;TAB 6;"NO.3 I
RON";TAB 18;8;TAB 22;"NO.8 IRON"

9600 PRINT TAB 2;4;TAB 6;"NO.4 I
RON";TAB 18;9;TAB 22;"SAND WEDGE
"
9610 PRINT TAB 2;5;TAB 6;"NO.5 I
RON";TAB 18;0;TAB 22;"PUTTER"
9620 PRINT
9630 PRINT "WE HAVE NINE HOLES F
OR YOU..... (DETAILS ON SCORE CA
RD LATER)"
9650 PRINT
9710 PRINT "PAR FOR COURSE IS 36
. GOOD LUCK.(PUSH ANY KEY TO STA
RT.)"

9730 PAUSE 4E4
9760 DIM S(9)
9770 LET N=0
9780 DIM D$(15,25)
9790 DIM Z(9)
9990 RETURN
```





Why wait any longer?

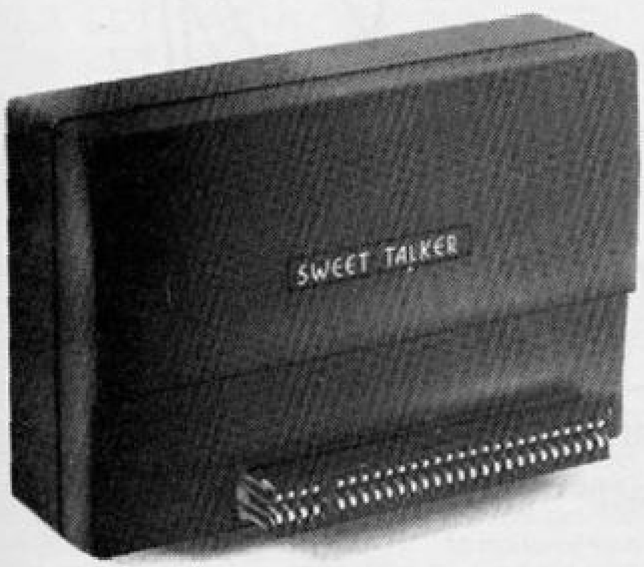
Upgrade your 16K ZX SPECTRUM Now!

The CHEETAH 32K RAMPACK simply plugs into the user port at the rear of your computer and increases the memory instantly to 48K.

- ★ Fully compatible with all SINCLAIR accessories via rear edge connector
- ★ **NO NEED TO OPEN COMPUTER AND INVALIDATE GUARANTEE**
- ★ Why send your computer away and wait weeks for upgrade
- ★ Fully cased tested and guaranteed.

only £39·95

including VAT and P&P.



Now make your Spectrum and ZX-81 Talk

Compatible with Interface I & II

The Cheetah "SWEET TALKER" just plugs into the back of the computer using the existing power supply. Based on an allophone system you can easily program any word sentence or phrase. Fully cased, tested guaranteed and compatible with all SINCLAIR accessories via rear edge connector. Complete with demonstration cassette and full instructions.
No more lonely nights!

Simply incredible at **£29·75**

(Please quote when ordering whether Spectrum or ZX81 owner)

16K RAM Pack for ZX-81	£19.75
64K RAM Pack for ZX-81	£44.75

Prices include VAT, postage & packing. Delivery normally 14 days. Export orders at no extra cost.

Dealer enquiries welcome

Cheetah, products available from branches of

John Menzies Boots WHSMITH Rumbelows

and all good computer stores

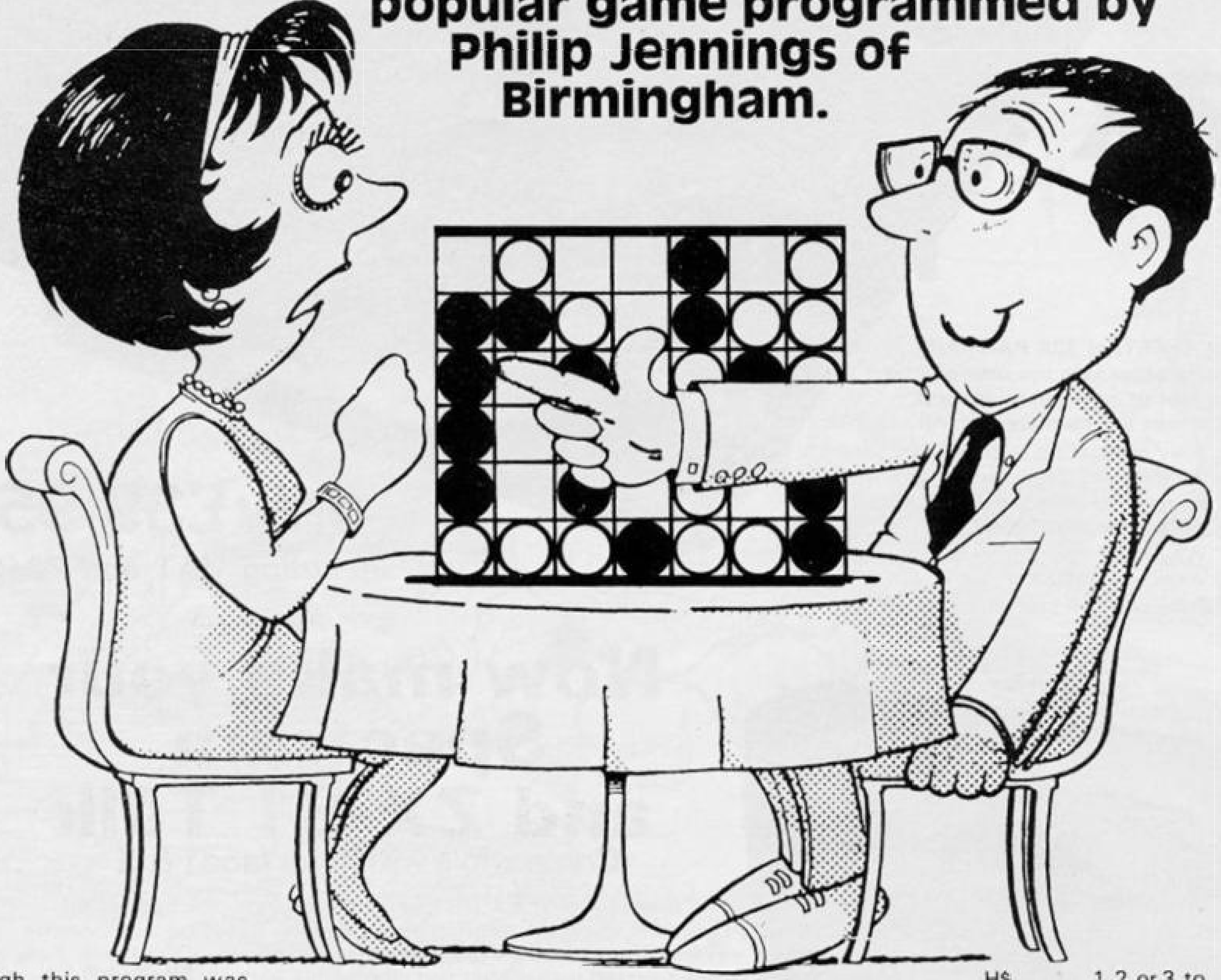
Cheetah Marketing

Cheetah Marketing Ltd, 24 Ray Street,
London EC1R 3DJ. Tel: 01 833 4909.
Telex: 8954958.

DEPT ZX

Connections

Play against the computer in this version of the popular game programmed by Philip Jennings of Birmingham.



Although this program was designed for the expanded ZX81, with only a slight modification the program will run comfortably on a Spectrum.

The game is based on the commercially available game for two players taking alternate goes in an attempt to get four of their own counters in a row, and at the same time trying to prevent the opponent achieving the same! However, being a computer, the ZX81 can go one better. Now you can play with no other opponent but your faithful ZX friend, and even choose one of three difficulty levels. If you insist on being conventional, the program also has the standard two player version built in.

After loading, you are asked for the number of players. To this you must respond by entering 1 or 2. If 1 is pressed, you are asked to choose whether or not you wish to go first, before being offered the choice of the three difficulty levels. This, incidently, is where the main game loop returns to, so that the difficulty level can be altered before each game. The playing area is printed on the screen, 6 rows of 7 columns, labelled A — G. Selecting one of the columns will cause your piece, either a 0 I or depending on whose go it is, to fall into place. Your object — to achieve a line, either vertical, diagonal or horizontal, of four your own pieces before your opponent can do the same. Facilities are included for recording the number of wins for each player, and as an added bonus, if you are playing against the computer, and it is your go, by pressing 0 (zero), the computer will take your go and play against itself!

Program notes

N$	Dimensioned array holding every combination of winning line.
Y$	The value of which, either 1 or 2 determines whose turn it is.
S1, S2	The scores for the players.
A-F	Represent the numbers 1-6. Included not so much for memory savers but to help the program run just a little bit faster.
P$	The number of players, 1 or 2
S$	Used in setting up initially to determine who is going to start.
H$	1,2 or 3 to determine the difficulty level.
A$ to G$	These hold the positions of pieces in the columns A to G.
J,K	General purpose.
I$	The selected column, A to G or 0.
L$	The playing piece.
J$	The selected column string is copied into J$, the piece held in L$ is added and J$ is returned to the selected column string.
Z	The first number of J$ which is not a space.
R	Simply change value to indicate a winning line.
Q$	Holds lines for comparison to assist the computer to

make its move.

Q — A for-next loop which governs how thoroughly the computer checks. The upper limit of Q is determined by H$.

The seven columns, A to G, are stored in A$ to G$, these originally consist of 6 spaces and an inverse space. The value of Y$ and P$ is checked to determine who plays. A check is made to ensure that not all columns are full, (ie a draw).

After a column is selected, a check is made to see if the 0 key was selected, in which case the program jumps straight into the computer playing routine.

If a column is chosen, the top of the column is checked to make sure that it is empty. The information already contained in the column is copied into J$, adjusted so that it contains the additional piece, and re-copied into the original string. With this additional piece, all 69 possible winning lines are re-entered in N$ This is checked for a winning line, and if found, R assumes the value of 1. If no winning line is found then Y$ assumes a new value, and the FOR-NEXT Q loop is visited to decide the best move for the computer. If there are two players, this loop is not visited.

Playing Hint

The computers priority is its own winning line. If it cannot find one it will endeavour to block your potential winning line, thus you can only win by tactically getting two potential winning lines in the same go. Of course, you can trust to luck, as the computer will occasionally make a random move if it can find no other that it considers suitable.

```
  10 DIM N$(69,4)
  20 LET Y$="1"
  30 LET S1=0
  35 LET I$=""
  40 LET S2=S1
  50 LET A=1
  60 LET B=A+A
  70 LET C=B+A
  80 LET D=C+A
  90 LET E=D+A
 100 LET F=E+A
 110 PRINT "SELECT NUMBER OF PLA
YERS",,,"1 OR 2"
 120 LET P$=INKEY$
 130 IF P$="2" THEN GOTO 230
 140 IF P$<>"1" THEN GOTO 120
 150 PRINT ,,"DO YOU WANT TO STA
RT? (Y/N)"
 160 LET S$=INKEY$
 170 IF S$="Y" THEN GOTO 200
 180 IF S$<>"N" THEN GOTO 160
 190 LET Y$="2"
 200 PRINT ,,"SELECT LEVEL OF DI
FFICULTY",,,"1, 2 OR 3"
 205 SLOW
 210 LET H$=INKEY$
 220 IF H$<CHR$ 29 OR H$>CHR$ 31
 THEN GOTO 210
 230 LET A$="      ■"
 235 LET R=0
 240 LET B$=A$
 250 LET C$=A$
 260 LET D$=A$
 270 LET E$=A$
 280 LET F$=A$
 290 LET G$=A$
 300 CLS
 310 FOR J=C TO 16
 320 FOR K=8 TO 22 STEP B
 330 PRINT AT J,K;"▒"
 340 NEXT K
 350 NEXT J
 360 FOR J=D TO 16 STEP B
 370 FOR K=9 TO 21 STEP B
 380 PRINT AT J,K;"▒"
 390 NEXT K
 400 NEXT J
 410 PRINT TAB 9;"A B C D E F G"
;TAB 23;" OR 0" AND P$="1"
 420 PRINT AT 8,0;"PLAYER 1";TAB
 23;"COMPUTER" AND P$="1";"PLAYE
R 2" AND P$="2";TAB 4;S1;TAB 28;
S2
 430 IF Y$="1" THEN GOTO 460
 440 IF P$="2" THEN GOTO 480
 450 IF Y$="2" THEN GOTO 1860
 460 PRINT AT 20,0;"PLAYER ONE T
O PLAY"
 470 GOTO 490
 480 PRINT AT 20,0;"PLAYER TWO T
O PLAY"
 485 SLOW
 490 IF A$(A)<>" " AND B$(A)<>"
" AND C$(A)<>" " AND D$(A)<>" "
AND E$(A)<>" " AND F$(A)<>" " AN
D G$(A)<>" " THEN GOTO 4700
 495 IF P$="1" AND (K=1 OR Y$="2
") THEN GOTO 560
 500 PRINT AT 19,9;"SELECT COLUM
N"
 510 LET I$=INKEY$
 515 SLOW
 520 IF I$="0" AND P$<>"2" THEN
GOTO 1860
 530 IF I$<CHR$ 38 OR I$>CHR$ 44
 THEN GOTO 510
 540 GOSUB 560
 550 GOTO 510
 560 IF I$="A" AND A$(1)<>" " TH
EN RETURN
 570 IF I$="A" AND A$(1)<>" " TH
EN RETURN
 580 IF I$="B" AND B$(1)<>" " TH
EN RETURN
 590 IF I$="C" AND C$(1)<>" " TH
EN RETURN
 600 IF I$="D" AND D$(1)<>" " TH
EN RETURN
 610 IF I$="E" AND E$(1)<>" " TH
EN RETURN
 620 IF I$="F" AND F$(1)<>" " TH
EN RETURN
 630 IF I$="G" AND G$(1)<>" " TH
EN RETURN
 640 LET L$="O"
 650 IF Y$="1" THEN LET L$="*"
```

```
 660 IF I$="A" THEN LET J$=A$
 670 IF I$="B" THEN LET J$=B$
 680 IF I$="C" THEN LET J$=C$
 690 IF I$="D" THEN LET J$=D$
 700 IF I$="E" THEN LET J$=E$
 705 IF I$="F" THEN LET J$=F$
 710 IF I$="G" THEN LET J$=G$
 720 FOR Z=1 TO 7
 730 IF J$(Z)<>" " THEN GOTO 750

 740 NEXT Z
 750 LET J$(Z-1)=L$
 760 LET R=0
 770 IF Y$="1" OR (Y$="2" AND P$
="2") THEN GOTO 1590
 780 FAST
 790 IF I$="A" THEN LET A$=J$
 800 IF I$="B" THEN LET B$=J$
 810 IF I$="C" THEN LET C$=J$
 820 IF I$="D" THEN LET D$=J$
 830 IF I$="E" THEN LET E$=J$
 840 IF I$="F" THEN LET F$=J$
 850 IF I$="G" THEN LET G$=J$
 860 LET N$(1)=A$(A TO D)
 870 LET N$(2)=B$(A TO D)
 880 LET N$(3)=C$(A TO D)
 890 LET N$(4)=D$(A TO D)
 900 LET N$(5)=E$(A TO D)
 910 LET N$(6)=F$(A TO D)
 920 LET N$(7)=G$(A TO D)
 930 LET N$(8)=A$(B TO E)
 940 LET N$(9)=B$(B TO E)
 950 LET N$(10)=C$(B TO E)
 960 LET N$(11)=D$(B TO E)
 970 LET N$(12)=E$(B TO E)
 980 LET N$(13)=F$(B TO E)
 990 LET N$(14)=G$(B TO E)
1000 LET N$(15)=A$(C TO F)
1010 LET N$(16)=B$(C TO F)
1020 LET N$(17)=C$(C TO F)
1030 LET N$(18)=D$(C TO F)
1040 LET N$(19)=E$(C TO F)
1050 LET N$(20)=F$(C TO F)
1060 LET N$(21)=G$(C TO F)
1070 LET N$(22)=A$(A)+B$(A)+C$(A
)+D$(A)
1080 LET N$(23)=A$(B)+B$(B)+C$(B
)+D$(B)
1090 LET N$(24)=A$(C)+B$(C)+C$(C
)+D$(C)
1100 LET N$(25)=A$(D)+B$(D)+C$(D
)+D$(D)
1110 LET N$(26)=A$(E)+B$(E)+C$(E
)+D$(E)
1120 LET N$(27)=A$(F)+B$(F)+C$(F
)+D$(F)
1130 LET N$(28)=B$(A)+C$(A)+D$(A
)+E$(A)
1140 LET N$(29)=B$(B)+C$(B)+D$(B
)+E$(B)
1150 LET N$(30)=B$(C)+C$(C)+D$(C
)+E$(C)
1160 LET N$(31)=B$(D)+C$(D)+D$(D
)+E$(D)
1170 LET N$(32)=B$(E)+C$(E)+D$(E
)+E$(E)
1180 LET N$(33)=B$(F)+C$(F)+D$(F
)+E$(F)
1190 LET N$(34)=C$(A)+D$(A)+E$(A
)+F$(A)
1200 LET N$(35)=C$(B)+D$(B)+E$(B
)+F$(B)
1210 LET N$(36)=C$(C)+D$(C)+E$(C
)+F$(C)
1220 LET N$(37)=C$(D)+D$(D)+E$(D
)+F$(D)
1230 LET N$(38)=C$(E)+D$(E)+E$(E
)+F$(E)
1240 LET N$(39)=C$(F)+D$(F)+E$(F
)+F$(F)
1250 LET N$(40)=D$(A)+E$(A)+F$(A
)+G$(A)
1260 LET N$(41)=D$(B)+E$(B)+F$(B
)+G$(B)
1270 LET N$(42)=D$(C)+E$(C)+F$(C
)+G$(C)
1280 LET N$(43)=D$(D)+E$(D)+F$(D
)+G$(D)
1290 LET N$(44)=D$(E)+E$(E)+F$(E
)+G$(E)
1300 LET N$(45)=D$(F)+E$(F)+F$(F
)+G$(F)
1310 LET N$(46)=D$(A)+E$(B)+F$(C
)+G$(D)
1320 LET N$(47)=D$(B)+E$(C)+F$(D
)+G$(E)
1330 LET N$(48)=D$(C)+E$(D)+F$(E
)+G$(F)
1340 LET N$(49)=C$(A)+D$(B)+E$(C
)+F$(D)
1350 LET N$(50)=C$(B)+D$(C)+E$(D
)+F$(E)
1360 LET N$(51)=C$(C)+D$(D)+E$(E
)+F$(F)
1370 LET N$(52)=B$(A)+C$(B)+D$(C
)+E$(D)
1380 LET N$(53)=B$(B)+C$(C)+D$(D
)+E$(E)
1390 LET N$(54)=B$(C)+C$(D)+D$(E
)+E$(F)
1400 LET N$(55)=A$(A)+B$(B)+C$(C
)+D$(D)
1410 LET N$(56)=A$(B)+B$(C)+C$(D
)+D$(E)
1420 LET N$(57)=A$(C)+B$(D)+C$(E
)+D$(F)
1430 LET N$(58)=A$(D)+B$(C)+C$(B
)+D$(A)
```

```
1440 LET N$(59)=A$(E)+B$(D)+C$(C
)+D$(B)
1450 LET N$(60)=A$(F)+B$(E)+C$(D
)+D$(C)
1460 LET N$(61)=B$(D)+C$(C)+D$(B
)+E$(A)
1470 LET N$(62)=B$(E)+C$(D)+D$(C
)+E$(B)
1480 LET N$(63)=B$(F)+C$(E)+D$(D
)+E$(C)
1490 LET N$(64)=C$(D)+D$(C)+E$(B
)+F$(A)
1500 LET N$(65)=C$(E)+D$(D)+E$(C
)+F$(B)
1510 LET N$(66)=C$(F)+D$(E)+E$(D
)+F$(C)
1520 LET N$(67)=D$(D)+E$(C)+F$(B
)+G$(A)
1530 LET N$(68)=D$(E)+E$(D)+F$(C
)+G$(B)
1540 LET N$(69)=D$(F)+E$(E)+F$(D
)+G$(C)
1550 FOR J=1 TO 69
1560 IF N$(J)=L$+L$+L$+L$ THEN L
ET R=1
1570 NEXT J
1580 IF Y$="1" OR P$="2" THEN GO
TO 1800
1590 SLOW
1595 FOR J=1 TO 5
1596 NEXT J
1600 FOR I=1 TO Z-1
1610 PRINT AT I*2+1,7+(CODE I$-3
7)*2;" ";AT I*2+3,7+(CODE I$-37)
*2;L$
1620 NEXT I
1622 LET K=0
1624 FOR J=1 TO 5
1625 NEXT J
1630 IF Y$="1" THEN GOTO 780
1635 IF P$="2" THEN GOTO 780
1640 GOTO 1800
1650 SLOW
1655 IF Y$="1" THEN GOTO 1730
1660 PRINT AT 20,0;"COMPUTER WIN
S    "
1670 IF P$="2" THEN GOTO 1690
1680 GOTO 1700
1690 PRINT AT 20,0;"PLAYER TWO W
INS    "
1700 LET S2=S2+1
1710 LET Y$="1"
1720 GOTO 1760
1730 PRINT AT 20,0;"PLAYER ONE W
INS    "
1740 LET S1=S1+1
1750 LET Y$="2"
1760 FOR J=1 TO 100
1770 NEXT J
1775 CLS
1780 IF P$="2" THEN GOTO 230
1790 GOTO 200
1800 IF R=1 THEN GOTO 1650
1810 IF Y$="1" THEN GOTO 1840
1820 LET Y$="1"
1830 GOTO 430
1840 LET Y$="2"
1850 GOTO 430
1860 DIM Q$(4,4)
1865 IF I$="0" THEN LET K=1
1870 FAST
1880 FOR Q=1 TO VAL H$*2
1890 IF I$="0" THEN GOTO 1930
1900 IF Q=1 OR Q=4 OR Q=6 THEN L
ET L$="O"
1910 IF Q=2 OR Q=3 OR Q=5 THEN L
ET L$="*"
1920 GOTO 1950
1930 IF Q=1 OR Q=4 OR Q=6 THEN L
ET L$="*"
1940 IF Q=2 OR Q=3 OR Q=5 THEN L
ET L$="O"
1950 IF Q>2 THEN GOTO 2050
2010 LET Q$(A)=" "+L$+L$+L$
2020 LET Q$(B)=L$+" "+L$+L$
2030 LET Q$(C)=L$+L$+" "+L$
2040 LET Q$(D)=L$+L$+L$+" "
2050 IF Q<3 THEN GOTO 2140
2060 LET Q$(A)=" "+L$+" "+L$
2070 LET Q$(B)=L$+" "+L$+" "
2080 LET Q$(C)=L$+L$+" "+" "
2085 LET Q$(D)=" "+L$+L$+" "
2090 IF Q<5 THEN GOTO 2140
2100 LET Q$(A)=" "+L$+" "+" "
2110 LET Q$(B)=" "+" "+L$+" "
2120 LET Q$(C)=" "+" "+" "+L$
2130 LET Q$(D)=" "+" "+L$+" "
2140 IF N$(A)=Q$(A) OR N$(8)=Q$(
A) OR N$(15)=Q$(A) THEN GOTO 400
0
2150 IF A$(B)<>" " AND (N$(22)=Q
$(A) OR N$(55)=Q$(A)) THEN GOTO
4000
2160 IF A$(C)<>" " AND (N$(23)=Q
$(A) OR N$(56)=Q$(A)) THEN GOTO
4000
2170 IF A$(D)<>" " AND (N$(24)=Q
$(A) OR N$(57)=Q$(A)) THEN GOTO
4000
2180 IF A$(E)<>" " AND (N$(25)=Q
$(A) OR N$(58)=Q$(A)) THEN GOTO
4000
2190 IF A$(F)<>" " AND (N$(26)=Q
$(A) OR N$(59)=Q$(A)) THEN GOTO
4000
2200 IF N$(27)=Q$(A) OR N$(60)=Q
$(A) THEN GOTO 4000
2210 IF N$(B)=Q$(A) OR N$(9)=Q$(
```

```
A) OR N$(16)=Q$(A) THEN GOTO 410
0
2220 IF B$(B)<>" " AND (N$(28)=Q
$(A) OR N$(52)=Q$(A)) THEN GOTO
4100
2230 IF B$(C)<>" " AND (N$(29)=Q
$(A) OR N$(53)=Q$(A)) THEN GOTO
4100
2240 IF B$(D)<>" " AND (N$(30)=Q
$(A) OR N$(54)=Q$(A)) THEN GOTO
4100
2250 IF B$(E)<>" " AND (N$(31)=Q
$(A) OR N$(61)=Q$(A)) THEN GOTO
4100
2260 IF B$(F)<>" " AND (N$(32)=Q
$(A) OR N$(62)=Q$(A)) THEN GOTO
4100
2270 IF N$(33)=Q$(A) OR N$(63)=Q
$(A) THEN GOTO 4100
2280 IF N$(C)=Q$(A) OR N$(10)=Q$
(A) OR N$(17)=Q$(A) THEN GOTO 42
00
2290 IF C$(B)<>" " AND (N$(34)=Q
$(A) OR N$(49)=Q$(A)) THEN GOTO
4200
2300 IF C$(C)<>" " AND (N$(35)=Q
$(A) OR N$(50)=Q$(A)) THEN GOTO
4200
2310 IF C$(D)<>" " AND (N$(36)=Q
$(A) OR N$(51)=Q$(A)) THEN GOTO
4200
2320 IF C$(E)<>" " AND (N$(37)=Q
$(A) OR N$(64)=Q$(A)) THEN GOTO
4200
2330 IF C$(F)<>" " AND (N$(38)=Q
$(A) OR N$(65)=Q$(A)) THEN GOTO
4200
2340 IF N$(39)=Q$(A) OR N$(66)=Q
$(A) THEN GOTO 4200
2350 IF N$(D)=Q$(A) OR N$(11)=Q$
(A) OR N$(18)=Q$(A) THEN GOTO 43
00
2360 IF D$(B)<>" " AND (N$(40)=Q
$(A) OR N$(46)=Q$(A)) THEN GOTO
4300
2370 IF D$(C)<>" " AND (N$(41)=Q
$(A) OR N$(47)=Q$(A)) THEN GOTO
4300
2380 IF D$(D)<>" " AND (N$(42)=Q
$(A) OR N$(48)=Q$(A)) THEN GOTO
4300
2390 IF D$(E)<>" " AND (N$(43)=Q
$(A) OR N$(67)=Q$(A)) THEN GOTO
4300
2400 IF D$(F)<>" " AND (N$(44)=Q
$(A) OR N$(68)=Q$(A)) THEN GOTO
4300
2410 IF N$(45)=Q$(A) OR N$(69)=Q
$(A) THEN GOTO 4300
2420 IF N$(E)=Q$(A) OR N$(12)=Q$
(A) OR N$(19)=Q$(A) THEN GOTO 44
00
2430 IF N$(F)=Q$(A) OR N$(13)=Q$
(A) OR N$(20)=Q$(A) THEN GOTO 45
00
2440 IF N$(7)=Q$(A) OR N$(14)=Q$
(A) OR N$(21)=Q$(A) THEN GOTO 46
00
2450 IF B$(B)<>" " AND (N$(22)=Q
$(B) OR N$(55)=Q$(B)) THEN GOTO
4100
2460 IF B$(C)<>" " AND (N$(23)=Q
$(B) OR N$(55)=Q$(B)) THEN GOTO
4100
2470 IF B$(D)<>" " AND (N$(24)=Q
$(B) OR N$(56)=Q$(B)) THEN GOTO
4100
2480 IF B$(E)<>" " AND (N$(25)=Q
$(B) OR N$(57)=Q$(B)) THEN GOTO
4100
2490 IF B$(F)<>" " AND (N$(26)=Q
$(B) OR N$(60)=Q$(B)) THEN GOTO
4100
2500 IF N$(27)=Q$(B) THEN GOTO 4
100
2510 IF C$(B)<>" " AND N$(28)=Q$
(B) THEN GOTO 4200
2520 IF C$(C)<>" " AND (N$(29)=Q
$(B) OR N$(52)=Q$(B)) THEN GOTO
4200
2530 IF C$(D)<>" " AND (N$(30)=Q
$(B) OR N$(61)=Q$(B)) THEN GOTO
4200
2540 IF C$(E)<>" " AND (N$(31)=Q
$(B) OR N$(62)=Q$(B)) THEN GOTO
4200
2550 IF C$(F)<>" " AND (N$(32)=Q
$(B) OR N$(63)=Q$(B)) THEN GOTO
4200
2560 IF N$(33)=Q$(B) THEN GOTO 4
200
2570 IF D$(B)<>" " AND N$(34)=Q$
(B) THEN GOTO 4300
2580 IF D$(C)<>" " AND (N$(35)=Q
$(B) OR N$(49)=Q$(B)) THEN GOTO
4300
2590 IF D$(D)<>" " AND (N$(36)=Q
$(B) OR N$(50)=Q$(B) OR N$(64)=Q
$(B)) THEN GOTO 4300
2600 IF D$(E)<>" " AND (N$(37)=Q
$(B) OR N$(51)=Q$(B) OR N$(65)=Q
$(B)) THEN GOTO 4300
2610 IF D$(F)<>" " AND (N$(38)=Q
$(B) OR N$(66)=Q$(B)) THEN GOTO
4300
2620 IF N$(39)=Q$(B) THEN GOTO 4
300
2630 IF E$(B)<>" " AND N$(40)=Q$
```

```
(B) THEN GOTO 4400
2640 IF E$(C)<>" " AND (N$(41)=Q
$(B) OR N$(46)=Q$(B)) THEN GOTO
4400
2650 IF E$(D)<>" " AND (N$(42)=Q
$(B) OR N$(47)=Q$(B) OR N$(67)=Q
$(B)) THEN GOTO 4400
2660 IF E$(E)<>" " AND (N$(43)=Q
$(B) OR N$(48)=Q$(B) OR N$(68)=Q
$(B)) THEN GOTO 4400
2670 IF E$(F)<>" " AND (N$(44)=Q
$(B) OR N$(69)=Q$(B)) THEN GOTO
4400
2680 IF N$(45)=Q$(B) THEN GOTO 4
400
2690 IF C$(B)<>" " AND N$(22)=Q$
(C) THEN GOTO 4200
2700 IF C$(C)<>" " AND (N$(23)=Q
$(C) OR N$(58)=Q$(C)) THEN GOTO
4200
2710 IF C$(D)<>" " AND (N$(24)=Q
$(C) OR N$(59)=Q$(C)) THEN GOTO
4200
2720 IF C$(E)<>" " AND (N$(25)=Q
$(C) OR N$(56)=Q$(C) OR N$(60)=Q
$(C)) THEN GOTO 4200
2730 IF C$(F)<>" " AND (N$(26)=Q
$(C) OR N$(57)=Q$(C)) THEN GOTO
4200
2740 IF N$(27)=Q$(C) THEN GOTO 4
200
2750 IF D$(B)<>" " AND N$(28)=Q$
(C) THEN GOTO 4300
2760 IF D$(C)<>" " AND (N$(29)=Q
$(C) OR N$(61)=Q$(C)) THEN GOTO
4300
2770 IF D$(D)<>" " AND (N$(30)=Q
$(C) OR N$(52)=Q$(C) OR N$(62)=Q
$(C)) THEN GOTO 4300
2780 IF D$(E)<>" " AND (N$(31)=Q
$(C) OR N$(53)=Q$(C) OR N$(63)=Q
$(C)) THEN GOTO 4300
2790 IF D$(F)<>" " AND (N$(32)=Q
$(C) OR N$(54)=Q$(C)) THEN GOTO
4300
2800 IF N$(33)=Q$(C) THEN GOTO 4
300
2810 IF E$(B)<>" " AND N$(34)=Q$
(C) THEN GOTO 4400
2820 IF E$(C)<>" " AND (N$(35)=Q
$(C) OR N$(64)=Q$(C)) THEN GOTO
4400
2830 IF E$(D)<>" " AND (N$(36)=Q
$(C) OR N$(49)=Q$(C) OR N$(65)=Q
$(C)) THEN GOTO 4400
2840 IF E$(E)<>" " AND (N$(37)=Q
$(C) OR N$(50)=Q$(C) OR N$(66)=Q
$(C)) THEN GOTO 4400
2850 IF E$(F)<>" " AND (N$(38)=Q
$(C) OR N$(51)=Q$(C)) THEN GOTO
4400
2860 IF N$(39)=Q$(C) THEN GOTO 4
400
2870 IF F$(B)<>" " AND N$(40)=Q$
(C) THEN GOTO 4500
2880 IF F$(C)<>" " AND (N$(41)=Q
$(C) OR N$(67)=Q$(C)) THEN GOTO
4500
2890 IF F$(D)<>" " AND (N$(42)=Q
$(C) OR N$(46)=Q$(C) OR N$(68)=Q
$(C)) THEN GOTO 4500
2900 IF F$(E)<>" " AND (N$(43)=Q
$(C) OR N$(47)=Q$(C) OR N$(69)=Q
$(C)) THEN GOTO 4500
2910 IF F$(F)<>" " AND (N$(44)=Q
$(C) OR N$(48)=Q$(C)) THEN GOTO
4500
2920 IF N$(45)=Q$(C) THEN GOTO 4
500
2930 IF D$(B)<>" " AND (N$(22)=Q
$(D) OR N$(58)=Q$(D)) THEN GOTO
4300
2940 IF D$(C)<>" " AND (N$(23)=Q
$(D) OR N$(59)=Q$(D)) THEN GOTO
4300
2950 IF D$(D)<>" " AND (N$(24)=Q
$(D) OR N$(60)=Q$(D)) THEN GOTO
4300
2960 IF D$(E)<>" " AND (N$(25)=Q
$(D) OR N$(55)=Q$(D)) THEN GOTO
4300
2970 IF D$(F)<>" " AND (N$(26)=Q
$(D) OR N$(56)=Q$(D)) THEN GOTO
4300
2980 IF N$(27)=Q$(D) OR N$(57)=Q
$(D) THEN GOTO 4300
2990 IF E$(B)<>" " AND (N$(28)=Q
$(D) OR N$(61)=Q$(D)) THEN GOTO
4300? ▞?
3000 IF E$(C)<>" " AND (N$(29)=Q
$(D) OR N$(62)=Q$(D)) THEN GOTO
4400
3010 IF E$(D)<>" " AND (N$(30)=Q
$(D) OR N$(63)=Q$(D)) THEN GOTO
4400
3020 IF E$(E)<>" " AND (N$(31)=Q
$(D) OR N$(52)=Q$(D)) THEN GOTO
4400
3030 IF E$(F)<>" " AND (N$(32)=Q
$(D) OR N$(53)=Q$(D)) THEN GOTO
4400
3040 IF N$(33)=Q$(D) OR N$(54)=Q
$(D) THEN GOTO 4400
3050 IF F$(B)<>" " AND (N$(34)=Q
$(D) OR N$(64)=Q$(D)) THEN GOTO
4500
3060 IF F$(C)<>" " AND (N$(35)=Q
$(D) OR N$(65)=Q$(D)) THEN GOTO
```

```
4500
3070 IF F$(D)<>" " AND (N$(36)=Q
$(D) OR N$(66)=Q$(D)) THEN GOTO
4500
3080 IF F$(E)<>" " AND (N$(37)=Q
$(D) OR N$(49)=Q$(D)) THEN GOTO
4500
3090 IF F$(F)<>" " AND (N$(38)=Q
$(D) OR N$(50)=Q$(D)) THEN GOTO
4500
3100 IF N$(39)=Q$(D) OR N$(51)=Q
$(D) THEN GOTO 4500
3890 IF G$(B)<>" " AND (N$(40)=Q
$(D) OR N$(67)=Q$(D)) THEN GOTO
4600
3900 IF G$(C)<>" " AND (N$(41)=Q
$(D) OR N$(68)=Q$(D)) THEN GOTO
4600
3910 IF G$(D)<>" " AND (N$(42)=Q
$(D) OR N$(69)=Q$(D)) THEN GOTO
4600
3920 IF G$(E)<>" " AND (N$(43)=Q
$(D) OR N$(46)=Q$(D)) THEN GOTO
4600
3930 IF G$(F)<>" " AND (N$(44)=Q
$(D) OR N$(47)=Q$(D)) THEN GOTO
4600
3940 IF N$(45)=Q$(D) OR N$(48)=Q
$(D) THEN GOTO 4600
3950 NEXT Q
3960 LET T=INT (RND*7)
3970 GOTO 4000+100*T
4000 LET I$="A"
4010 GOSUB 490
4100 LET I$="B"
4110 GOSUB 490
4200 LET I$="C"
4210 GOSUB 490
4300 LET I$="D"
4310 GOSUB 490
4400 LET I$="E"
4410 GOSUB 490
4500 LET I$="F"
4510 GOSUB 490
4600 LET I$="G"
4610 GOSUB 490
4620 GOTO 3960
4700 PRINT AT 20,0;"A DRAW
     "
4705 SLOW
4710 GOTO 1760
9000 CLEAR
9010 SAVE "ZX-CONNECT  "
9020 GOTO 1
```

TWO GREATS FOR THE SINCLAIR QL



Tim Hartnell's QL Games Compendium

From Tim Hartnell (the author, PCW magazine called 'Mr Sinclair'), comes this major games book for the QL . . .

Not only does it contain listings for more than 20 complete games, but the detailed instructions and ideas will help you develop your own games, whilst improving your programming as you do so. There is even a complete chapter to teach you how to create moving graphics games on your QL!

The Programs include:

ELIZA — in which your QL acts as a psychiatrist and talks to you.

OTHELLO — challenge your QL to a round of Reversi/Othello.

MAGICIAN'S MAZE — high adventure exploring ruins after gold.

DETECTIVE Q — solve the murder; different each time it runs.

STOCKBROKER — a challenging simulation puts thousands in your control as you work towards getting a million.

TRYST WITH MEPHISTOPHELES — nearly 20K of challenging adventure.

QL RACER — real-time moving graphics, in a game few will win.

Exploring the Sinclair QL
An Introduction to SuperBASIC

Here's your chance to explore the potential of SuperBASIC, the sophisticated language supplied with the Sinclair QL.

Designed to help you write better, clearer programs, Sinclair SuperBASIC is a state-of-the-art programming language, a language which keeps pace with the hardware innovations of the computer.

Andrew Nelson, author of several popular computer books, explains all the important concepts, key words, commands and statements of SuperBASIC. In this book, he explains that many of the programming skills you have from the Spectrum can be directly transferred to the QL, and show how — as you learn the new words in the QL's vocabulary — your programs can gradually be made more complex and sophisticated, by incorporating the new commands and statements.

Interface Publications, Department QZX
9-11 Kensington High Street, London W8 5NP.

Please send me the following books.
I enclose a total of £ ______

☐ **Tim Hartnell's QL Games Compendium** — Tim Hartnell, ISBN 0 947695 04 4	£5.95
☐ **Exploring the Sinclair QL — An Introduction to SuperBASIC** Andrew Nelson, ISBN 0 907563 84 8 —	£4.95
☐ **Mastering Machine Code on your ZX Spectrum** — Toni Baker	£9.95
☐ **Spectrum Machine Code Made Easy, Volume One for beginners** — James Walsh	£5.95
☐ **Spectrum Machine Code Made Easy, Volume Two for advanced programmers** — Paul Holmes	£5.95
☐ **The Spectrum Machine Code Reference Guide** — Richard Ross-Langley	£4.95
☐ **Creating Adventures on your ZX Spectrum** — Peter Shaw and James Mortleman	£4.95
☐ **Putting Your Spectrum to Work (including 15 major programs, a word processor and spread-sheet calculations)** — Chris Callender	£4.95
☐ **Creating Arcade Games on the ZX Spectrum** — Daniel Haywood	£3.95
☐ **Programming Your ZX Spectrum** — Hartnell/Jones	£6.95
☐ **60 Games and Applications for the ZX Spectrum** — David Harwood	£4.95
☐ **Beyond Simple BASIC — Delving Deeper into your ZX Spectrum** — Dilwyn Jones	£7.95
☐ **Instant Spectrum Programming (tape and book)** — Tim Hartnell	£4.95
☐ **20 Simple Electronic Projects for the Spectrum** — Stephen Adams	£6.45

Interface Publications, Department QZX
9-11 Kensington High Street, London W8 5NP.

Name ....

Address ....

....

....

(TRADE ONLY: Interface Publications are distributed exclusively in the UK and Eire by WHS Distributors. Export trade [illegible] by Interface Publications.)

INTERFACE PUBLICATIONS

We're the experts!

Across The Pond

This issue our man in the States, Mark Fendrick, compares the Spectrum and the TIS2068.

Since this is a column in a British magazine, written by an American author about an American computer with British roots (follow that?), it is only proper that we devote this month's column to a comparison of the ZX Spectrum, and the American version known as the Timex/Sinclair 2068. Quicksilva (who have an office here in Texas), and Melbourne House have made Spectrum software available already, and Richard Shepherd Software (who have supplied me with many excellent titles for review in my U.S. column) should follow soon. There seems to be hope for us yet!

While Timex's modem was finally released by the third party manufacturer who was to produce it for Timex, the biggest disappointment has been the lack of microdrives. Many of us originally purchased our T/S 2068's because of the promise of microdrives. A few disk drive interfaces are in the works (as are other goodies), but still we wait.

As always, I look forward to your INPUT, and welcome any comments, tips, or what have you. Write to me at;

Changes

The first and most noticeable change, is in the case and keyboard of the T/S 2068. The original T/S 2000 was in the balck case with rubber "chicklet" style keys. While this was an improvement over the T/S 1000 (ZX-81), Timex changed the dimensions of the case to 14 ½" × 7 ½" × 1". The case now became silver/grey, and contained a hinged door at the right top which would be used for cartridge based software. (More about this later.) They keys were replaced by a full size keyboard with 42 solid plastic, soft touch keys, including a full size space bar. (One drawback to this keyboard is that the space bar seems to be "dead" on the ends, and must be hit close to the centre.) The sides now contained Atari standard joystick ports, which are supported by a new STICK command in BASIC. The final noticeable change is the addition of an on/off switch on the left side of the computer.

At first, Timex had planned on marketing two different versions — the 16K T/S 2016 ($149.95), and the 48K T/S 2048 ($199.95). Sound familiar? In the end, however, only the 48K model was released, dubbed the Timex/Sinclair 2068. Actually, it is in reality a 72K unit, containing a 24K ROM (16K+8K bank switched to handle the cassette interface), and a 48K RAM. It is these ROM additions and changes that account for the incompatibilities that exist. As a rule, most software written in BASIC will LOAD and RUN properly, while machine code software will not. Additionally, the positioning of the individual bus lines in the edge connector are not in the same order as on the Spectrum, causing further, hardware incompatibility.

BASIC Additions

Changes to the Spectrum BASIC came about as well. In order to utilize the joystick ports, Timex added a command — STICK. This command reads the condition of the switches in the attached joystick. The form of the command would be as such;

IF STICK (1,2)=1 THEN LET X=X+1

The number 1 in the parenthesis defines either the button (1) or the stick (2). The second number refers to the left (1) or right (2) joystick. The T/S 2068 can read 9 different positions from the joystick. This opens up many possible uses of the joystick in a number of programs.

A second set of additions are error trapping features. These work with the ON ERR command. ON ERR GO TO xx place into a program supresses the error reports, and causes the program to jump to line xx when any error is encountered. ON ERR RESET causes the error reports to be reinstituted, and ON ERR CONTinue, causes the program to go back to the line where the error took place. These are very powerful commands to prevent a program from stopping when an error occurs, and can also be used to protect a program from being stopped (BREAK is considered an error) and listed.

The final addition to BASIC is an enhanced version of the BEEP command (which is also available on the T/S 2068), called appropriately enough — SOUND. The T/S 2068 has a three channel sound system, and each channel is controlled by fifteen registers. These registers control the pitch, duration, and volume of the sound being produced by that channel. You also have control of the *envelope* or shape of the sound. This means that using the SOUND command, and a lot of work, a T/S 2068 can play music in three part harmony. This command can also be used to produce a wide range of sound effects, and the manual illustrates gunshots, an explosion, and a whistling bomb. An interesting command, but very difficult to use. None of the programmers that I know (myself included) have been able to make much use of this facility. (There is one program named MUSICOLA which not only takes advantage of this capability, but allows the user the chance to compose music with harmony without understanding the SOUND command.)

Cartridge Software

Earlier we spoke about a hinged door on the top of the case which was to be used for cartridge based software. Timex had planned to release most of their software in two forms; cassette, and **Command Cartridges**. These Command Cartridges would utilize the 'chip on a board' technology, whereby the IC would be attached directly to the board, eliminating the IC holder, and producing a very thin (just over 0.5 inch thick) wafer. This board would be encased in a 2.75 inch ⅹ2.5 inch case to be inserted into the Command Cartridge port. When the computer is turned on, and senses the presence of a cartridge in this port, it automatically bank switches (the T/S 2068 contains bank switching architecture allowing the use of up to 16 Mb!) to use (up to) an extra 56K.

However, as we here in North America are all too painfully aware, Timex withdrew from the computer marketplace before many of these command cartridges were produced, and since a great number of these cartridges must be ordered for production, no new Command Cartridges seem to be in the works from any supplier. Those of use who have a cartridge or two are fairly lucky. (I own two — States & Capitols, and Casino I.)

This leads up to a current use of this port which is most encouraging for T/S 2068 owners. Since the majority of software for the Spectrum would not work on the T/S 2068, the introduction of a SPECTRUM EMULATOR was much anticipated and has finally become a reality. Originally developed by Timex during the development of the T/S 2068, I was shown one originally by Dan Ross (former Vice President of Timex) at a Timex Celebration in Boston. Had Timex continued with computers, this would have been marketed. When Timex backed off, the founder of a Timex users group in North Carolina made some contacts, did some research, and produced an emulator to be inserted into the Command Cartridge port, and introduce a pseudo Spectrum ROM. With this board in place, a T/S 2068 will run almost all of the software). Also available are chops which get placed inside the case (permanently) to give the same capabilities. One company has developed a method whereby both the Sinclair and Timex ROMs coexist, and can be selected via a magnetic switch. (Note: The T/S 2068 is not made to be opened by the user as is the SPECTRUM, since no upgrades were available or planned. Americans in general seem to be less inclined to tinker than our British cousins.)

Great Selection

The great selection of software that has been available to you in the U.K. is now starting to become available in the U.S.

Mark L. Fendrick
P.O. Box 2392
Secaucus, New Jersey
07094-0992
U.S.A.

Until next time . . .

Spectrum Arcade

David Harwood, our latest addition to the ZXC team, continues his quest for the perfect arcade game

Some large enterprises seem now to have cottoned onto the large software boom, the amount of profit that can be made and that software is the in thing.

K-Tel are venturing into software and should have new titles out in time for christmas (such are the publication dates that although I am writing this article just before Christmas, you will not read it until the new year) and also an interesting company going into software is British Telecom. Do you get a reduction in the software price if you have shares in the company.

Software till now has always been in the price range £5-£10 with a few retailing at £15. British Telecom could be the company that brings the price down to a more realistic figure, greatly easing the pocket. One would presume that with cheap software, the sales would be higher, but I wonder if they would be so high as to equal the same profit if they have sold them at £5.75.

Crazy Caverns

The first of the BT software is Crazy Caverns, which fits into the 16K Spectrum (some companies have not forgotten you, 16K users).

It is interesting that the introductory screen is not your usual screen loaded from the tape, but is drawn from the loader program, producing a swirling graphic display before the main program is loaded.

A 16K cassette is noticeably faster to load than a 40K one, which is a great plus for 16K games as I find it extremely non-productive waiting for games to load.

Either using a cursor joystick or keys O, A, O and P, the idea is to go round a screen (again!) and feed the mouth with the Crunchy Cubes that are positioned around the screen. The idea is to line up the Crunchy Cubes and shoot them into the muching mouth.

Yes, there are Crazy Crashers which move in an annoyingly random way to prevent you accomplishing your task.

The idea of this game is innovative, but boredom ensues as it is not very exciting just going round screens as feeder, shoving biscuits into a chomping mouth.

This game would have been better a year ago, as it appears to be rather dated now, but for the 16K Spectrum, what can you expect?

INSTRUCTIONS	–	**70%**
PRESENTATION	–	**75%**
ADDICTABILITY	–	**75%**
VALUE FOR MONEY	–	**95%**
ZXC FACTOR	–	**7**

Exodus

The next one from BT is EXODUS and before I start I must mention the excellent introduction before each game is played.

The word EXODUS falls onto the screen, a letter at a time, with a clear clicky laser sound and the border along with each letter containing bright flashing lines in unison with the sound. It is a pity that this screen does not occur more often as I could have watched this for ages, much better than the washing machine.

Keys can be re-defined with a selection of joysticks available, but with your own key definition this game is compatible with any type of joystick.

A nice edition also is the instruction mode, which prevents the usual fumbling for the cassette in order to read the tiny printed directions.

The idea of the game is as follows. There is a pit, obviously in the outer galactic sphere and you need to stop the countless objects from climbing to the outside by shooting them with your laser. The pit is in the shape of a hexagon and your ship can either move clockwise or anticlockwise around the boundary of the pit.

If an object manages to reach the rim of a pit, you must not touch it as a life will be lost. There are also other handicaps. If a mutant llama manages to escape (I didn't quite manage to catch sight of any mutant bits) and reach the outside rim, another life will be lost. You have also got a friend Spud (as in potato) and if you happen to shoot him you will lose a life as well as a friend!!!

There too are various advantages, like a bonus of 1000 every 10,000 points and a time bonus for shooting a clock will add time to your limit of 79 seconds.

As well as the vast scoring scheme for many different and interesting objects, if you fire and hit a Television that is tuned to Channel 4, you score an extra 20 points as this is so rare (A wonderful idea and true!!!)

The game is fun to play, the only real difficulty is to stop the mutant llamas escaping and not to get Spud. As long as you stay clear from any object leaving the pit, you should not have too many problems.

I first of all wondered why the pit was in the shape of a hexagon, but I soon found out that it was to bring over a false sense of speed. Travelling around the pit is fine, but if you want to move quick in order to reach a llama, your instinct has got to be quick as the shape slows you down.

I did not quite understand the time limit as when the time reaches 0, a life was lost instead of perhaps going onto the next level. There might be a target number of objects before they stop jumping from the pit, but if this game has got any faults, this is the only one.

Although the graphics are not very interesting, they are clear and with lively sound this game is a sure winner for the mere price of £2.50. Should keep you amused for ages.

INSTRUCTIONS	–	**80%**
PRESENTATION	–	**75%**
ADDICTABILITY	–	**75%**
VALUE FOR MONEY	–	**99%**
ZXC FACTOR	–	**8**

Cyclon Attack

A and F software (using the motto 'Nulli Secundus', which for you non-classicists stands for 'Secnd To None') have produced a 3D Space Battle for the 48K Spectrum called New Cyclon Attack.

The claim that it has insometric perspective in 3D is perhaps questionable, but the game is played in 3-Dimensional graphics and is of the Star Wars standard.

The Cyclons are sending wave after wave of attacking fleets. Each time a new wave appears, you are launched from your Mother ship and after each wave has been destroyed, you need to dock in order to refuel.

There are various instruments positioned at the top of the screen, detailed as follows. A radar, giving full information about the approaching wave attack, a shield strength gauge (there is an automatic safety shield should you come under fire, but this obviously deteriorates in strength depending on use), laser status and fuel status.

There is a sight in the centre of the screen with climb and turn indicators, enabling you to position the laser beam to any position on the screen in the attempt to clear each Cyclon wave.

Some nice added touches, the ability to save/load the high score table, the ability to redefine the keys and detailed screens with satellites, planets, shooting stars etc. all go forward to make Cyclon Attack a first class arcade game with excellent graphics and realistic sound.

INSTRUCTIONS	–	**95%**
PRESENTATION	–	**100%**
ADDICTABILITY	–	**95%**
VALUE FOR MONEY	–	**90%**
ZXC FACTOR	–	**9**

Kung Fu

I cannot say a lot about this Kung-Fu game from Bug-Byte software, as the idea is very simple. You are a Kung-Fooist and have to fight your opponent until he drops down (dead? I don't know) for you to win.

Please forgive the lack of technical terms, but there are four moves at your fingertips, the left chop, the forward punch, the right kick and the left kick with forward and backward movement. The keys used are 1, 2, 3, 4, break and sym shift respectively for player one with the opposite keys for player 2. ▶

There is joystick compatibility with the Kempston, but the moves are quite easy to master on a keyboard.

The scenario is obviously Chinese style, with Orientated music, which can be very annoying, so I was grateful for the music on/off facility.

There are two different options to play either against the Spectrum or another player. For some reason, if you fought against the Spectrum, 3 rounds were attempted inside a Chinese Style house, but if against another player, only one round outside was played.

The replay function is quite a good idea, enabling you to watch the tense action all over again and for the enthusiast, a ranking is given telling you the colour of your belt.

A novel game of skill, including excellent animated graphic sequences (nulli secundus). A bit of a no-goer for me, as there is not a lot of action.

INSTRUCTIONS	–	**70%**
PRESENTATION	–	**95%**
ADDICTABILITY	–	**70%**
VALUE FOR MONEY	–	**75%**
ZXC FACTOR	–	**7**

Sir Lancelot

Another game for the 16K Spectrum (hooray!) is Sir Lancelot from Melbourne House, so a good quality game can be expected.

Sir Lancelot is loaded using rapidload, which is supposed to speed up loading times by half. This is a very good idea for 48K games (why didn't they adopt this with Holmes?), but there is no real need for this with a 16K game and I did not notice any improvement in loading speed.

Unfortunately, this is another grab and collect program and is of the same genre as all the others. Can we possibly have something else please?

You, as Sir Lancelot, need to go through 24 rooms in the castle and collect all the treasure that is hanging around. Animals do the guarding of the treasure and there is a time limit for you to gather everything in the room and scarper to the exit before the time runs out and a life is lost.

The action is fast, as you move Sir Lancelot around with O for left, P for right and CAP SHIFT to jump.

With a suggested completion time of three weeks, the game should keep you amused for some time. The graphics and sound are alright, but nothing brilliant. The largest letdown is that every time a life is lost, you have to wait for the funeral march to be played.

INSTRUCTIONS	–	**70%**
PRESENTATION	–	**70%**
ADDICTABILITY	–	**70%**
VALUE FOR MONEY	–	**75%**
ZXC FACTOR		**7**

Hyperaction

Now we come onto Hyperaction, a game of intrigue and wits for the 48K Spectrum. Produced by Silversoft, we should expect an arcade quality game, but do we have it?

First of all, a minus. In these days of first impressions etc, etc, etc, an introductory screen is very important. It not only makes the waiting for the program to load acceptable, but it also impresses the player with a taste of the game to follow.

This game has got various screens, although I only managed to progress to the second, but perhaps those with nimble fingers or joysticks should get onto to more.

You are a spider with only 6 legs, I suppose the Spectrum's graphics handling made the decision on the last two and are faced initially with a grid, which turns into a random maze after the computer has knocked out various holes.

In the first screen there are acorns and mushrooms along the path of the maze with the letters ZX hanging around the screen. Your task is to gather all the ZX's in order to get to the next level. Pacmen have been sent in order to prevent you from doing this and they have been well trained in their mission.

You can change the paths of the maze by forcing the acorns to form different paths, but you need to be careful not to force an acorn onto a ZX or a Pacman as you will lose a life.

The first screen is very hard to master at first, but once you get the idea it is very hard to stop playing. You need to design your own way of approach, but I decided to block out the opposing pacmen in the first screen, by moving the wall and then gathering the ZX's.

The same idea is on screen 2, but there are Arctic Jellyfish instead of Pacmen and this time you have to cover all the blue ground on the screen (ice), but as you are still in the maze, you can move walls etc.

If the first two screens are anything to go by, the rest should be very good, I am still trying to get onto screen 3.

There is quite a bad point in that between each new attempt, there is a delay while you are introduced to Pacmen, Jellyfish etc, and the name of the next screen, but this should not hinder your excitement.

Hyperaction is an excellent and original arcade game, with Sinclair, Cursor and Kempston joystick compatibility.

INSTRUCTIONS	–	**65%**
PRESENTATION	–	**75%**
ADDICTABILITY	–	**85%**
VALUE FOR MONEY	–	**80%**
ZXC FACTOR	–	**8**

Eddie Kidd Jump Challenge

Fancy yourself as a stunt rider? The dare-devil feat can now be achieved within the safety and comfort of your own home. the Eddie Kidd Jump Challenge is a new game for the 48K Spectrum, putting you in the driver seat of your own motorbike.



You are Eddie Kidd, who we are told is the most naturally gifted motorcycle stuntman in the world and are privileged to take the Jump Challenge to gain a world record.

Initially, before jumping over cars with a motorcycle you have to jump over barrels with a BMX bike. Having successfully done this, you graduate to the motorcycle, with the controls of speed, brakes and position of rider all at hand in order to give you a realistic approach of riding a bike.

When you have left the ramp with enough room for your run up, you can start.

I cannot really say much about this game as it totally relies on the skill of the player in manoeuvrering the stunt bike.

Various points need to be considered while attempting a jump, eg. speed, length of run up, position on bike and the only way to make a successful one is to get a feel for the bike and learn by your mistakes.

I, not a bike rider at heart, did find it quite difficult to complete a successful motorbike jump and correctly judge the above three factors.

The game starts off with 8 cars and progresses onwards a car at a time. Unfortunately you always return to the BMX jump after each crash, which is a bit annoying as this jump is the only one that is easy to master. It would have been an idea to be able to attempt any level of jump at any time.

The company Martech are also putting on a competition, with BMX bikes, colour TV's, MSX Computers (not Spectrum Plus?) for prizes and runners up getting a poster of Eddie Kidd (WOW!). A good idea which should increase sales if not stunt riders.

A skillful game with excellent graphics and realistic sound all add up to a winner.

INSTRUCTIONS	–	**90%**
PRESENTATION	–	**90%**
ADDICTABILITY	–	**85%**
VALUE FOR MONEY	–	**80%**
ZXC FACTOR	–	**8**

CRAZY CAVERNS AND EXODUS:
FIREBIRD SOFTWARE,
WELLINGTON HOUSE
UPPER ST. MARTIN'S LANE,
LONDON WC2H 9DL
PRICE: £2.50

CYCLON ATTACK:
A & F SOFTWARE LTD.,
UNIT 8
CANAL SIDE INDUSTRIAL ESTATE,
WOODBINE STREET EAST,
ROCHDALE,
LANCASHIRE.
PRICE: TBA

KUNG-FU:
BUG-BYTE,
98-100 THE ALBANY,
OLD HALL STREET,
LIVERPOOL.
PRICE: TBA

SIR LANCELOT:
MELBOURNE HOUSE PUBLISHERS
39, MILON TRADING ESTATE,
ABINGDON,
OXON.
PRICE: £6.95

EDDIE KIDD JUMP CHALLENGE:
MARTECH,
MARTECH HOUSE,
BAY TERRACE,
PEVENSEY BAY,
EAST SUSSEX.
PRICE: £6.95

POWERFUL AND INEXPENSIVE BUSINESS SOFTWARE FOR ZX81, T/S1000 and T/S1500 COMPUTERS

ZX-TEXT



A word processor is to a computer user what a typewriter is to a typist, except that the former has more advantages than the latter. ZX-Text can operate in 16-64K RAM providing from 1350 to 9000 words per document. It features 6 different options: write, read, edit, print, save and clear text. Text is written on a per-line basis with quick speed and with horizontal back-space and delete capabilities being available. You can also access the editor directly from write mode and vice-versa. Text can be proof-read on a per-line basis allowing for enough time to determine if any editing is needed. The text editor allows a line of text to be deleted, inserted, replaced and listed for editing. You may also change a word or expression within a line, stop or start text while it is scrolling up the screen, begin reading text from the first line of the file, re-enter write mode from the editor, return to the main-menu or create a window so that you can read-edit two files simultaneously. The print option takes text displayed in 30-column format on the screen and outputs to either the ZX/TS printer. (With Memotech's Centronics Parallel Interface 80-column and lower/higher - case output is possible.) Files may be saved on tape cassette with the use of one single command, or by the same token they can be erased from memory / storage so that the full capacity of the program can be used for other purposes such as composing letters, reports, articles, memos, standard forms, instructions, ads, graphs, telephone directory, lists of customers, members, friends...etc. Also copies of files are always less expensive and easier to run than using a photocopier. Other advantages are savings in time, paper, ink, correcting mistakes and adding afterthoughts more efficiently than doing them through either handwriting or using a typewriter.

$11.95

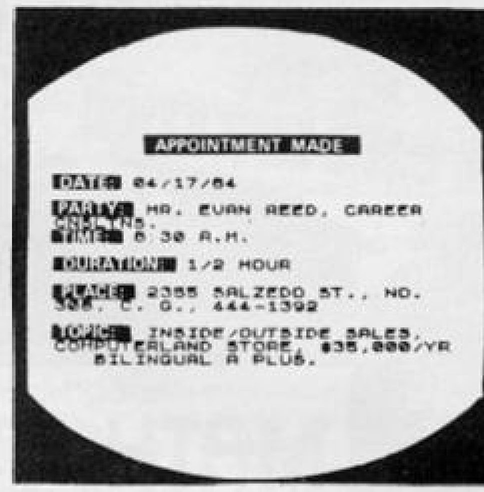
ZX-CALC



An electronic spreadsheet calculator is the fundamental basic tool for summarising, reporting and analyzing in matrix form any accounting, mathematical or scientific manipulation of numbers. ZX-Calc operates in 32-64K RAM and affords a maximum of 3360 characters / spreadsheet. The entire matrix consists of 15 columns (letters A-O) and 30 rows (numbers 1-30) with 8 characters / cell. Unlike other popular ESCs, ZX-Calc uses in calculations and within cells all 14 math functions on the ZX-81 / TS1000. It offers a unique "SUM function that totals one or more rows / columns simultaneously. Parenthesis can be used within equations. There is no fixed limit on how many equations may be entered. Formulas may be stored in all 420 cells of the spreadsheet. The display affords 15 rows / colums. Loading of data into more than one cell can occur across / down one or more row / column simultaneously. With vertical windowing you can arrange a set of columns in any order, or practice using fixed-variable-alignment display formats. The menu offers 6 options: enter / erase, move, calculate, print, save and clear the spreadsheet. Enter / erase allows the entering, deletion or data alignment within a cell through the use of a mobile cursor. With the move option you may move around the entire sreadsheet to access any row, column or cell. The calculate option allows you to enter labels, values or formulas into a cell or write and enter equations that will act upon the data already within the spreadsheet. You can also enter bar graphs into a cell in this option. Absolute / relative replication, down / across a column / row, is also allowed by this option. Also this option allows the automatic calculation of the entire spreadsheet with one single command. Print allows you to output to either the ZX / TS printer the entire spreadsheet by column-sets and row-pages through use of the COPY command. The entire spreadsheet may be saved on cassette tape or you may clear all data from it or erase the program from RAM entirely. The most salient advantage provided by an ESC over specifically vertical applications software is that an ESC provides a reusable framework with which you can compose any specific financial model rather than just be limited to only one statically fixed format for storing, displaying and manipulating numerical data.

$11.95

ZX-CALENDAR

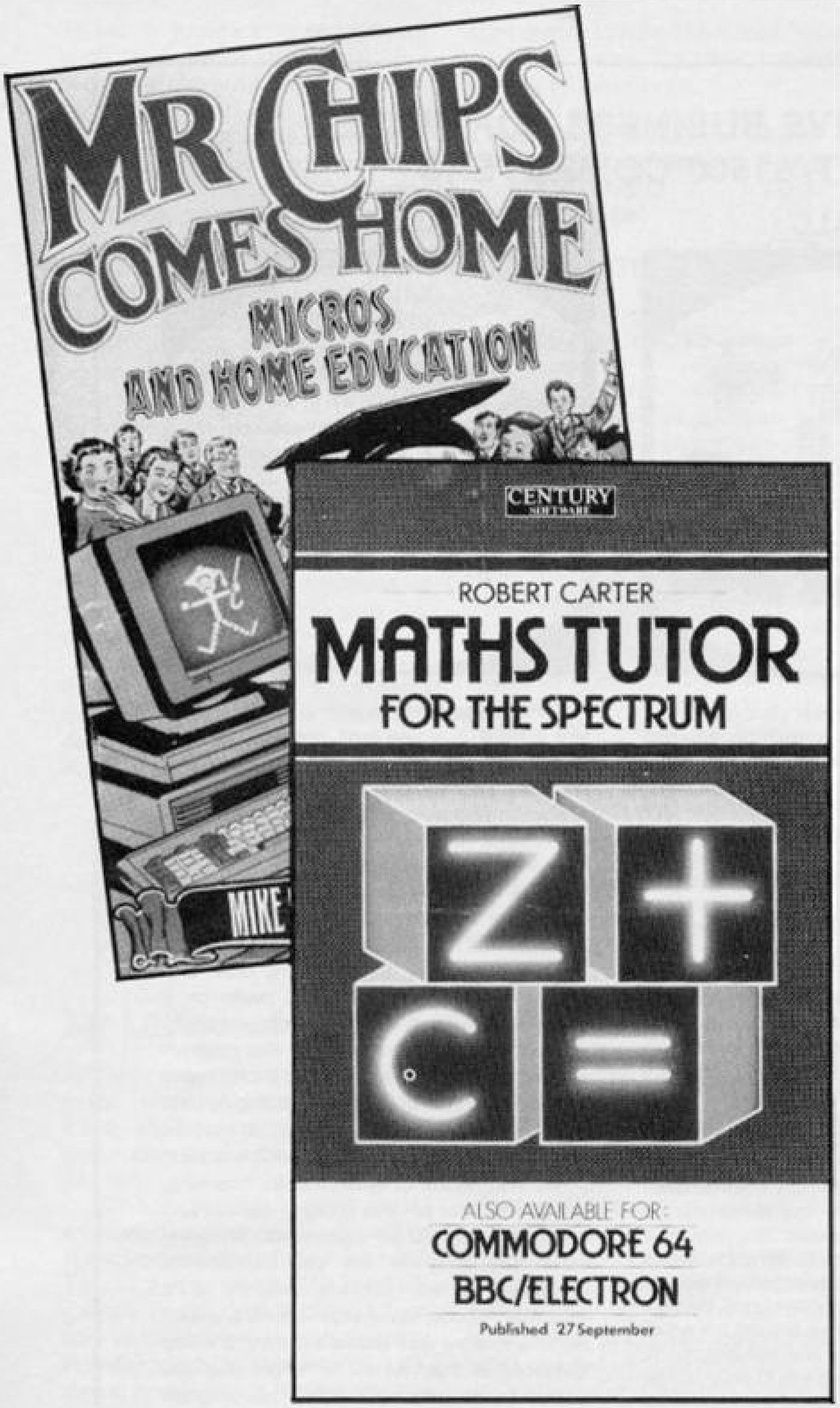


Time management is an important aspect of any serious business and personal agenda. Planning how to spend our time leaves us better prepared before and while we are spending it and we remain better organized after we finish spending it. ZX-Calendar operates in 16-64K RAM affording 30 appointments in 16K, 100 in 32K, 180 in 48K or 250 in 64K. Each appointment record holds a maximum of 220 characters. The main menu includes enter, search/check/sort, change, save, clear and print any and all appointments made on a specific date or with any party. Output to either the ZX/TS printer is permissible. This program will permit you to remember to do something or to be somewhere important by cataloging your answers to six questions that you must account for in order not to waste time when it is scarce: when, with whom, at what time, for how long, where and what are you going to discuss and conclude when you get together with someone else? The program lets you permanently originate, record, classify, search, sort, calculate, modify, summarize, obtain a written report and store your answers to the preceding questions so that you will not forget what you decide to do with your time. This program identifies your time according to when you are going to spend it and with whom you are going to share it. Through these forms of labeling appointments you are able to verify or modify how your time is budgeted without wasting ink, paper or more time trying to remember what you said to yourself or what someone else said to you or where you placed certain written messages that you now can't find. With this program you will know where you can find exactly what you need to know about where you want to and have to be, or where you have been, before you get and after you got there. Thus, ZX-Calendar will let you plan your time so that you will never have to worry about what is ahead or what came before, for you will always know, by using it, to never be caught astray by any time-frame.

$11.95

$1.50 SHIPPING AND HANDLING / PROGRAM

A.F.R. SOFTWARE - 1605 Pennsylvania Avenue, No. 204 - Miami Beach, Florida 33139
(305) 531-6464



BOOK NEWS

Maths Tutor from Century Communications provides an introduction to mathematics for the non-mathematical person and aims above all to make maths fun. Readers do not have to have extensive knowledge of the Sinclair Spectrum; the first chapter (optional) is especially for those who know absolutely nothing about the machine. It is meant for all people who have access to the Spectrum; those who want to learn maths and are looking for the simplest way in which to do it; games players who are looking for alternative uses for their machines; or those of any age who just want to brush-up their knowledge.

The book could also be used in the classroom by teachers who want to illustrate mathematical principles and techniques — most O-level and A-level syllabus material is covered.

A cassette containing the programs listed in this book is available and the book and cassette have been combned as a software pack.

In . . . . Brief

• **Print and Plotter Products,** the company who produce the excellent Jotters for the ZX81 and Spectrum, now bring us **Adventure Planner.**

This is a 50 page planning pad containing a mapping system with over 150 locations and includes examples and tips.

I must admit that this is always the area of playing adventure games which causes me the most problems. At £3.95 I will undoubtedly invest in one of these!

• For all brain strain fanatics "**The Times Book of Computer Puzzles & Games for the Sinclair Spectrum**" (how do they get it all on the cover ?) at £6.95 provides a selection of program listings sent in by programmers hoping to win one of the prizes they offered.

• **MR CHIPS COMES HOME**: Micros and Home Education by Bryan Skinner and Mike Gerrard (£7.95) is a complete guide to using a micro for education at home. The book includes a comprehensive software review section divided by machine. Bryan Skinner, a former teacher, is software editor of Personal Computer News, and Mike Gerrard is a regular contributor to Which Micro? and Personal Computer News.

• Look out for the distinctive, colourful and large "**step-by-step**" series of computer programming tutors from **Dorling Kindersley Publishers Ltd.**

Two for the common Spectrum and four for the Spectrum + and all at £5.95 each. Great for the absolute beginner.

In fact Sinclair Research were so impressed they got them to write and produce the Spectrum + manual (also available separately for £4.95).

• **Papermac** have published two rather specialist tomes, "**How to buy Software**" by Alfred Glossbrenner and "**Women and Computing: The golden opportunity**" by Rose Deakin.

• For those upgrading or have bought a QL, a comprehensive series of books have been published by **Hutchinson Computer Publishing Co Ltd.**

These have been produced with the hlp and co-operation of Sinclair Research and Psion and are priced at £6.95 each. The titles are self explanatory and are:

Introducing the Sinclair QL
Introduction to Super Basic on the Sinclair QL
Advanced programming with the Sinclair QL
Word Processing with the Sinclair QL
Desk Top Computing with the Sinclair QL

Hutchinson are one of the first companies to produce something for this machine and a quick glance gives the impression that they have not been rushed onto the market but are very well written.

• Superbasic is from Granada and has been written by AA Berk. Certainly worth looking for at £6.95. Also from them is **QL Computing** by the prolific **Ian Sinclair,** an author of great experience. A well written book and at £5.95 is one of the few, but increasing, books available for this machine.

• **Pan** and **Personal Computer News** have published a book for an increasng market, "**25 Programs for the Sinclair ZX Microdrive: Multi-user Games for the Spectrum**".

Available for £5.95 and, interestingly, not for sale in Canada!

• **Ian Stewart** takes a jokey approach to computing in his book "**Gateway to Computing with the ZX Spectrum**" books 1 and 2. Golly Jennings, aren't computer book titles getting length?! The Editor's typing finger must be getting sore!

Anyway, its published by **Shiva** for £4.95 and its big, bright and amusing, helps put the fun back into computing.

MAIL ORDER ADVERTISING

British Code of Advertising Practice

ADVERTISEMENTS IN THIS PUBLICATION ARE REQUIRED TO CONFORM TO THE BRITISH CODE OF ADVERTISING PRACTICE. IN RESPECT OF MAIL ORDER ADVERTISEMENTS WHERE MONEY IS PAID IN ADVANCE, THE CODE REQUIRES ADVERTISERS TO FULFIL ORDERS WITHIN 28 DAYS, UNLESS A LONGER DELIVERY PERIOD IS STATED. WHERE GOODS ARE RETURNED UNDAMAGED WITHIN SEVEN DAYS, THE PURCHASER'S MONEY MUST BE REFUNDED. PLEASE RETAIN PROOF OF POSTAGE/DESPATCH, AS THIS MAY BE NEEDED.

Mail Order Protection Scheme

If you order goods from Mail Order advertisements in this magazine and pay by post in advance of delivery, *Movie Maker* will consider you for compensation if the Advertiser should become insolvent or bankrupt, provided:

1. You have not received the goods or had your money returned; and
2. You write to the Publisher of this publication; summarising the situation not earlier than 28 days from the day you sent your order and not later than two months from that day.

Please do not wait until the last moment to inform us. When you write, we will tell you how to make your claim and what evidence of payment is required.

We guarantee to meet claims from readers made in accordance with the above procedure as soon as possible after the Advertiser has been declared bankrupt or insolvent (up to a limit of £2,000 per annum for any one Advertiser so affected and up to £6,000 per annum in respect of all insolvent Advertisers. Claims may be paid for higher amounts, or when the above procedure has not been complied with, at the discretion of this publication, but we do not guarantee to do so in view of the need to set some limit to this commitment and to learn quickly of readers' difficulties.)

This guarantee covers only advance payment sent in direct response to an advertisement in this magazine (not, for example, payment made in response to catalogues etc, received as a result of answering such advertisements). Classified advertisements are excluded.


An Argus Specialist Publication
YOUR BEST BUYER'S GUIDE TO SOFTWARE
Feburary/March 1985 £1.95
SOFTWARE
TODAY
LADDERS AND PLATFORMS—A STEP BY STEP GUIDE
DEADLY OR DANGEROUS—CAN YOU MASTER ACORNSOFT'S
Out Now
PLUS: PLUS: PLUS: PLUS: PLUS:
BBC, SPECTRUM, AMSTRAD,
CBM 64, TI-99/4A AND
DRAGON SOFTWARE

GENCALC

A brilliant program from David F M Carmichael of Glasgow

This program has been devised so that the ZX Spectrum and ZX printer can be used to carry out and annotate a wide range of calculations without the necessity of having to write specialised prorams for each case. The program operates in a number of different modes which can be selected by codes entered at the key board.

The main operating modes (with their appropriate codes) are:-

"**type**": Typewriting mode allowing full typewriting facilities in columns 1-30 of the display with various delete options.
"**calc**": Calculator mode allowing typewriting facilities similar to the above but only in columns 1-15 while allowing the right hand side of the screen to be used as a calculator. Easy interchange is possible between the typing and calculating sub-modes.
"**table**": User defined formula driven table calculator which can be used with complex arithmetical expressions with up to 3 variables. Optional total facility is available.

Two other modes are available:

"**copy**": Copies the screen and gives option to take a further copy if required.
"**newpage**": Copies the screen as above but on completion scrolls the display to the next page.

All the above codes are initialised in the program so that in the event of a crash or break in the program, it can be restarted without disturbing the display by using the command GOTO "(whatever code appropriate)"

On running the program the prompt "Enter required mode" will appear and the appropriate code should be entered from the keyboard. The computer will not accept spurious codes and will beep to indicate the fact. (Indeed all through the program spurious entries will get the same treatment with one notable exception in "table" mode).

Detailed instructions for each mode are as follows:

"type" : Flashing "T" cursor displayed initially. Simulates typewriter for all key entries (except for those below). A warning beep is given when the cursor reaches column 25 and a double beep at column 30. The cursor automatically moves on to the next line but there is no autorepeat for key entry.

Special key entries (with shift where appropriate) are:-

'DELETE'	: Autorepeat backspace/delete
'ENTER'	: Autorepeat new line
'EDIT'	: Gives "Enter required mode' prompt
'STOP'	: Stop program
'Cursor Left'	: Gives line delete cursor (flashing " ✱" in column 0).

With line delete-cursor the following key entries (with shift where appropriate) can be used.

Cursor Up **Cursor Down**	Moves cursor in appropriate direction autorepeat).
'DELETE'	Deletes whole line to the right of the cursor.
Cursor right	Returns to normal typewriter state with "T" cursor on column 1 of the line to the right of the cursor.

NOTE: When the bottom of the screen is reached either by normal typing or by use of the newline command the screen promt "Screen full — go to next page?" will appear. If the 'N' key is pressed this prompt will disappear andthe option of obtaining and using the delete cursor is open. If the 'Y' key is pressed an automatic copy/newpage sequence is initiated.

'note' sub-mode	Flashing 'N' cursor shows initially. Simulates typewriter as in "type" mode but only in left hand side of screen. The delete cursor is a flashing "+" and will only delete one half of the line. All control key entries are the same as for "type" mode with the following additions:
"N" cursor	Cursor right goes to 'calculator' sub-mode A flashing " " cursor will appear on the right of the screen marking the print line.
"+" cursor	Provided that no calculation is in progress the key entry "Cursor Left" will move the delete cursor (behind the screen as it were) to the right side of the screen. Here it can be used to delete any entries on that side. At this point "Cursor Right" will return the cursor to the left hand side of the screen. It is not possible to go to the calculator sub-mode directly with the "+" cursor on the right side.

'Calculator' sub mode

The prompt will ask for number or operator entry. Number of entries must be no more than 8 character long. Spurious entries will be beeped.

Relevant operators are:

"+"	for addition.
"−"	for subtraction
"✱"	for multiplication.
"/"	for division.
"+%"	calculates % and adds result from running total.
"−%"	calculates % and subtracts result to running total.
"sub"	gives subtotal (marked with symbol " ")
"total"	gives total.
"note"	returns to "note" sub-mode.

Notes

The calculator will only work if numbers are displayed in normal notation. Where subtotals or totals are such that the computer would require to use exponent notation for display then a series of beeps will occur and thetotal or subtotal will appear as a line of asterisks.

Automatic subtotalling takes place where this will clarify the display.

Where the screen is filled in the 'note' sub-mode the option to continue or to select the line delete cursor is given as for 'type'. However if the screen is filled in 'calculator' sub-mode the copy/newpage sequence is automatically initiated.

2, 3 or 4 columns can be chosen and after entering the no required the code letters for the columns are displayed ('a' 'b' 'c' and 'd' with the right hand column always being 'd'). Column descriptions up to seven characters long can now be systematically entered (and corrected if necessary).

The formula is now entered. This can be any arithmetical expression up to 30 characters long but must use the code letters of the columns. The computer will display "d =" and only the right hand side of the equation is entered.

The calculator is normally set to give a max of six decimal places but the opportunity is now given to alter this.

Entry prompts will now appear systematically (using the column descriptions already entered. Entries with more than seven characters will be refused (and beeped). Any non numeric entry will also be beeped but will be accepted (for reasons noted below). The opportunity is given to check all entries before computation of the value is column (d) is carried out. Non numerica entries are acceptable provided the column is not being included on the calculation (eg. if formula is d = b ✱ c then column (a) can be used as columns used for computation will result in an error. As in "calc" any total in column (d) outwith normal numeric display will be printed as asterisks (with appropriate beeps).

After each line the opportunity is given to continue the calculation or to stop and if the latter the option of a total of column (d) is given. Unlike the intermediate line totals the total of column (d) can be displayed in either normal or exponent notation and thus can be any number within the range of the computer. On completion of all calculations the computer displays the 'Enter required mode' prompt.

NOTE: If on first selecting the "table" mode the print cursor is below line 15 the computer will ▶

automatically give a new page. Subsequent automatic copy/newpage sequences will take place at line 19 and the current column headings will be displayed automatically at the top of the new page.

Technical notes

'Descriptive' variables are used throughout the program and most of these are self explanatory.

Various flags are also used and these are noted below:

- **notec** : Determines which prompt is given in 'calculator' submode of "calc" when changing from 'note' submode. (if 0 then 'number' is prompted : If 1 then 'operator' is prompted).
- **subtf** : Controls whether automatic subtotalling will take place on entry of certain operators. If 0 then no subtotalling takes place. The subtotal routine automatically resets this flag to 0.
- **calcf** : Controls whether automatic subtotalling takes place after calculation. In the case of addition + subtraction this remains 0 but other operators such as " * " or "/" set this to 1. Like subtf this is reset to 0 after subtotalling.
- **cursf** : Is set to 1 as soon as calculation starts and controls movement of line delete cursor to right hand side of screen (which can only happen when this flag = 0).
- **totf** : Determines whether total is in normal or exponent notation and sets print position accordingly.
- **nof** : Set at 1 by "no, check" routine (Line 9700) if entry is non numeric; otherwise 0.

Variables

The following non descriptive variables are used:

a$	
b$	General purpose non dimensioned.
d$ (4,7)	Column descriptions in 'table' mode.
e$ (4,7)	input information in 'table' mode.
a)	
b)	
c)	
d)	general purpose variables.
e)	
f)	
t	'table' total
x)	
y)	general print position controls
z	Special print position control used in 'table' mode (see line 4220 and similar).

Functions

Two used defined functions are included:

Line 9800	Controls no of decimal points.
Line 9810	% function used in 'calc' mode.

```
1 REM "GENCALC-1984"
1Ø CLS : PRINT AT 1,Ø;"Program loaded";AT 3,Ø;"Please stop recorder and wait"
2Ø PAUSE 1ØØ
3Ø LET no=Ø: LET yes=1: LET page=1: LET newpage=96ØØ: LET copy=93ØØ: LET copyok=941Ø: LET info=942Ø: LET scroll=943Ø: LET tableok=944Ø: LET colsok=945Ø: LET dpok=946Ø: LET nocheck=97ØØ
4Ø LET cursf=Ø: LET notec=Ø: LET dp=6: LET x=1: LET y=1: LET mode=5ØØ: LET type=1ØØØ: LET calc=2ØØØ: LET subt=34ØØ: LET table=4ØØØ
45 DIM f$(32)
5Ø CLS : PRINT AT 4,6;"GENERAL CALCULATION";AT 6,12;"PROGRAM";AT 8,2;" david fm carmichael 1984)"
6Ø PRINT AT 12,4;"Input instructions will";AT 13,4;"appear in"; FLASH 1;AT 13,14;"flashing"; FLASH Ø;AT 13,23;"mode";AT 15,4;"Other instructions and";AT 16,4;"questions in plain text"
7Ø PRINT AT 18,2;"Answer questions by pressing";AT 19,1;"'N' for 'NO' and 'Y' for 'YES'"
8Ø PRINT #1;TAB 3;;"To continue press any key": PAUSE Ø
9Ø CLS
4ØØ REM **page no**
41Ø LET x=1: LET y=1: PRINT AT Ø,24;"page ";page
5ØØ REM **mode"selection"
51Ø BEEP .Ø5,2Ø: INPUT  FLASH 1;"Enter required mode"; FLASH Ø;" "; LINE a$
52Ø IF a$="type" THEN  GO TO type
53Ø IF a$="calc" THEN  GO TO calc
54Ø IF a$="table" THEN  GO TO table
55Ø IF a$="copy" THEN  PRINT AT x,y; FLASH Ø;" ": GO SUB copy
56Ø IF a$="newpage" THEN  GO SUB newpage
57Ø REM **space for other modes**
99Ø BEEP .1,1Ø: GO TO 5ØØ
1ØØØ REM **type mode**
1Ø1Ø PRINT AT x,y; FLASH 1;"T": PAUSE Ø
1Ø2Ø LET a$=INKEY$: IF a$="" THEN  GO TO 1Ø2Ø
1Ø3Ø BEEP .ØØ5,2
11ØØ REM **controls**
111Ø IF a$=CHR$ 8 THEN  GO TO 15ØØ
112Ø IF a$=CHR$ 12 THEN  GO TO 14ØØ
113Ø IF a$=CHR$ 13 THEN  GO TO 13ØØ
114Ø IF a$=CHR$ 7 THEN  GO TO mode
119Ø IF a$=CHR$ 226 THEN  STOP
12ØØ REM **type**
121Ø IF y=3Ø AND x<21 THEN  PRINT AT x,y;a$;AT x+1,1; FLASH 1;"T": GO TO 123Ø
1215 IF y=31 AND x=21 THEN  GO T
```

```
O 1Ø2Ø
122Ø PRINT AT x,y;a$;AT x,y+1; F
LASH 1;"T"
123Ø LET y=y+1
124Ø IF y=25 THEN  BEEP .Ø5,15
125Ø IF y=3Ø THEN  BEEP .Ø5,15:
PAUSE 1Ø: BEEP .Ø5,15
126Ø IF y=31 AND x<21 THEN  LET
y=1: LET x=x+1: GO TO 128Ø
127Ø IF y=31 AND x=21 THEN  BEEP
 1,15: GO SUB scroll: IF ans=yes
 THEN  GO SUB newpage: PRINT AT
x,y; FLASH 1;"T"
128Ø LET b$=INKEY$: IF b$=a$ THE
N  GO TO 128Ø
129Ø GO TO 1Ø2Ø
13ØØ REM **newline**
131Ø BEEP .Ø5,15
132Ø IF x=21 THEN  PRINT AT x,y;
" ": LET y=31: PRINT AT x,y; FLA
SH 1;"T": GO TO 127Ø
133Ø PRINT AT x,y; FLASH Ø;" ";A
T x+1,1; FLASH 1;"T"
134Ø LET x=x+1: LET y=1
135Ø PAUSE 5
136Ø GO TO 1Ø2Ø
14ØØ REM **backspace/delete**
141Ø IF y=1 AND x>1 THEN  PRINT
AT x,1;" ";AT x-1,3Ø; FLASH 1;"T
"
142Ø IF x=1 AND y=1 THEN  PRINT
AT x,y+1;" ";AT x,y; FLASH 1;"T"
: BEEP 1,15: GO TO 1Ø2Ø
143Ø IF y<>1 THEN  PRINT AT x,y;
" ";AT x,y-1; FLASH 1;"T"
144Ø LET y=y-1
145Ø IF y=Ø THEN  LET y=3Ø: LET
x=x-1
146Ø IF x=Ø THEN  LET x=1: LET y
=1
147Ø BEEP .Ø5,1Ø
148Ø LET b$=INKEY$: IF b$=a$ THE
N  PAUSE Ø
149Ø GO TO 1Ø2Ø
15ØØ REM **delete /cursor**
151Ø PRINT AT x,y;" ";AT x,Ø; FL
ASH 1;"*"
152Ø LET b$=INKEY$: IF b$=a$ THE
N  GO TO 152Ø
153Ø LET a$=INKEY$: IF a$="" THE
N  GO TO 153Ø
154Ø IF a$=CHR$ 9 THEN  GO TO 18
5Ø
155Ø IF a$=CHR$ 1Ø THEN  GO TO 1
66Ø
156Ø IF a$=CHR$ 11 THEN  GO TO 1
6ØØ
157Ø IF a$=CHR$ 12 THEN  GO TO 1
8ØØ
158Ø LET b$=INKEY$: IF b$=a$ THE
N  GO TO 158Ø
159Ø GO TO 153Ø
16ØØ REM **cursor up**
161Ø IF x=1 THEN  GO TO 159Ø
162Ø PRINT AT x,Ø;" ";AT x-1,Ø;
FLASH 1;"*"
163Ø LET x=x-1
1635 BEEP .Ø5,25
164Ø LET b$=INKEY$: IF b$=a$ THE
N  PAUSE Ø
165Ø GO TO 153Ø
166Ø REM **cursor down**
167Ø IF x=21 THEN  GO TO 159Ø
168Ø PRINT AT x,Ø;" ";AT x+1,Ø;
FLASH 1;"*"
169Ø LET x=x+1
1695 BEEP .Ø5,25
17ØØ LET b$=INKEY$: IF b$=a$ THE
N  PAUSE Ø
171Ø GO TO 153Ø
18ØØ REM **line delete**
181Ø PRINT AT x,1;"
                    "
182Ø LET b$=INKEY$: IF b$=a$ THE
N  GO TO 182Ø
183Ø BEEP .1,1Ø
184Ø GO TO 153Ø
185Ø REM **clear delete/cursor**
186Ø PRINT AT x,Ø;" ";AT x,1; FL
ASH 1;"T"
1865 LET y=1
187Ø LET b$=INKEY$: IF b$=a$ THE
N  GO TO 187Ø
188Ø BEEP .1,1Ø
189Ø GO TO 1Ø2Ø
2ØØØ REM **calc mode**
2ØØ2 LET dp=6
2ØØ5 IF y>15 THEN  PRINT AT x,y;
" ": LET y=1: LET x=x+1
2Ø1Ø PRINT AT x,y; FLASH 1;"N":
PAUSE Ø
2Ø2Ø LET a$=INKEY$: IF a$="" THE
N  GO TO 2Ø2Ø
2Ø3Ø BEEP .ØØ5,2
21ØØ REM **controls**
211Ø IF a$=CHR$ 8 THEN  GO TO 25
ØØ
212Ø IF a$=CHR$ 12 THEN  GO TO 2
4ØØ
213Ø IF a$=CHR$ 13 THEN  GO TO 2
3ØØ
214Ø IF a$=CHR$ 7 THEN  GO TO mo
de
215Ø IF a$=CHR$ 9 AND notec=Ø TH
EN  GO TO 3ØØØ
2155 IF a$=CHR$ 9 AND notec=1 TH
EN  PRINT AT x,y;" ";AT x,31; FL
ASH 1;"<": GO TO 316Ø
```

```
2190 IF a$=CHR$ 226 THEN  STOP
2200 REM **note**
2210 IF y=15 AND x<21 THEN  PRIN
T AT x,y;a$;AT x+1,1; FLASH 1;"N
": GO TO 2230
2220 PRINT AT x,y;a$;AT x,y+1; F
LASH 1;"N"
2230 LET y=y+1
2240 IF y=12 THEN  BEEP .05,15
2250 IF y=15 THEN  BEEP .05,15:
PAUSE 10: BEEP .05,15
2260 IF y=16 AND x<21 THEN  LET
y=1: LET x=x+1
2270 IF y=16 AND x>=20 THEN  BEE
P 1,15: GO SUB scroll: IF ans=ye
s THEN  GO SUB newpage: PRINT AT
 x,y; FLASH 1;"N"
2280 LET b$=INKEY$: IF b$=a$ THE
N  GO TO 2280
2290 GO TO 2020
2300 REM **newline**
2310 BEEP .05,15
2320 IF x=20 THEN  PRINT AT x,y;
" ": LET y=16: PRINT AT x,y; FLA
SH 1;"N": GO TO 2270
2330 PRINT AT x,y;" ";AT x+1,1;
FLASH 1;"N"
2340 LET x=x+1: LET y=1
2350 PAUSE 5
2360 GO TO 2020
2400 REM **backspace/delete**
2410 IF y=1 AND x>1 THEN  PRINT
AT x,1;" ";AT x-1,15; FLASH 1;"N
"
2420 IF x=1 AND y=1 THEN  PRINT
AT x,y+1;" ";AT x,y; FLASH 1;"N"
: BEEP 1,15: GO TO 2020
2430 IF y<>1 THEN  PRINT AT x,y;
" ";AT x,y-1; FLASH 1;"N"
2440 LET y=y-1
2450 IF y=0 THEN  LET y=15: LET
x=x-1
2460 IF x=0 THEN  LET x=1: LET y
=1
2470 BEEP .05,10
2480 LET b$=INKEY$: IF b$=a$ THE
N  PAUSE 0
2490 GO TO 2020
2500 REM **delete /cursor**
2510 PRINT AT x,y;" ";AT x,0; FL
ASH 1;"+"
2520 LET b$=INKEY$: IF b$=a$ THE
N  GO TO 2520
2530 LET a$=INKEY$: IF a$="" THE
N  GO TO 2530
2540 IF a$=CHR$ 9 THEN  GO TO 28
50
2545 IF a$=CHR$ 8 AND cursf=0 TH
EN  PRINT AT x,0;" ": BEEP .1,25
: GO TO 3600
2550 IF a$=CHR$ 10 THEN  GO TO 2
660
2560 IF a$=CHR$ 11 THEN  GO TO 2
600
2570 IF a$=CHR$ 12 THEN  GO TO 2
800
2580 LET b$=INKEY$: IF b$=a$ THE
N  GO TO 2580
2590 GO TO 2530
2600 REM **cursor up**
2610 IF x=1 THEN  GO TO 2590
2620 PRINT AT x,0;" ";AT x-1,0;
FLASH 1;"+"
2630 LET x=x-1
2635 BEEP .05,25
2640 LET b$=INKEY$: IF b$=a$ THE
N  PAUSE 0
2650 GO TO 2530
2660 REM **cursor down**
2670 IF x=21 THEN  GO TO 2590
2680 PRINT AT x,0;" ";AT x+1,0;
FLASH 1;"+"
2690 LET x=x+1
2695 BEEP .05,25
2700 LET b$=INKEY$: IF b$=a$ THE
N  PAUSE 0
2710 GO TO 2530
2800 REM **line delete**
2810 PRINT AT x,1;"
 "
2820 LET b$=INKEY$: IF b$=a$ THE
N  GO TO 2820
2830 BEEP .1,10
2840 GO TO 2530
2850 REM **clear delete/cursor**
2860 PRINT AT x,0;" ";AT x,1; FL
ASH 1;"N"
2865 LET y=1
2870 LET b$=INKEY$: IF b$=a$ THE
N  GO TO 2870
2880 BEEP .1,10
2890 GO TO 2020
3000 REM **calculator**
3010 REM ** no. and mode **
3020 LET c$="": LET cursf=1: LET
 subtf=0: LET calcf=0
3030 PRINT AT x,y;" ": PRINT AT
x,31; FLASH 1;"<"
3035 REM ** number entry **
3040 INPUT  FLASH 1;"Enter numbe
r"; FLASH 0;" "; LINE a$: GO SUB
 nocheck: IF nof=1 THEN  BEEP .1
,15: GO TO 3040
3050 IF LEN a$>8 THEN  BEEP .2,2
0: BEEP .2,20: BEEP .2,20: BEEP
.2,20: GO TO 3040
3100 IF a$(1)<>"-" THEN  PRINT A
T x,22;a$
```

```
3105 IF a$(1)="-" THEN  PRINT AT
 x,21;a$
3130 LET c$=c$+a$
3140 PRINT AT x,31;" ": LET x=x+
1: PRINT AT x,31; FLASH 1;"<"
3145 IF x>=21 THEN  GO SUB newpa
ge
3147 IF calcf=1 THEN  GO SUB sub
t: LET subtf=0
3150 REM ** operator **
3160 INPUT  FLASH 1;"Enter opera
tor"; FLASH 0;" "; LINE b$
3170 IF b$="note" THEN  PRINT AT
 x,31;" ": LET notec=1: LET y=1:
 GO TO 2000
3180 IF b$="total" THEN  GO TO 3
500
3190 IF b$="sub" THEN  LET subtf
=1: GO SUB subt: GO TO 3160
3194 IF calcf=1 AND notec=1 THEN
  PRINT AT x,31;" ": LET x=x+1:
IF x>=21 THEN  GO SUB newpage
3198 IF calcf=1 AND notec=1 THEN
 GO SUB subt
3200 IF b$="+" THEN  PRINT AT x,
16;"add": LET subtf=1: GO TO 331
0
3210 IF b$="-" THEN  PRINT AT x,
16;"ddct": LET subtf=1: GO TO 33
10
3220 IF b$="*" THEN  GO SUB subt
: PRINT AT x,16;"mult": LET calc
f=1: LET subtf=1: GO TO 3310
3230 IF b$="/" THEN  GO SUB subt
: PRINT AT x,16;"div": LET calcf
=1: LET subtf=1: GO TO 3310
3235 IF x>=19 AND (b$="+%" OR b$
="-%") THEN  GO SUB newpage: LET
 x=1
3240 IF b$="+%" THEN  GO SUB sub
t: PRINT AT x,16;"add%"
3250 IF b$="-%" THEN  GO SUB sub
t: PRINT AT x,16;"ddct%"
3254 IF b$="+%" OR b$="-%" THEN
 LET subtf=0: GO TO 3260
3258 BEEP .1,10: GO TO 3160
3260 PRINT AT x,31; FLASH 0;" ":
 LET x=x+1: PRINT AT x,31; FLASH
 1;"<"
3265 INPUT  FLASH 1;"Enter perce
ntage"; FLASH 0;" "; LINE a$: IF
 LEN a$>5 THEN  BEEP .1,10: GO T
O 3265
3268 GO SUB nocheck: IF nof=1 TH
EN  BEEP .1,10: GO TO 3265
3270 PRINT AT x,16;a$
3275 LET f=VAL a$: LET a=VAL c$
3280 LET b=FN p(a)
3282 PRINT AT x,23;b;AT x,31; FL
ASH 0;" ": LET x=x+1: IF x=21 TH
EN  GO SUB newpage
3283 PRINT AT x,31; FLASH 1;"<"
3284 IF b$="-%" THEN  LET a=a-b
3286 IF b$="+%" THEN  LET a=a+b
3288 IF a<>0 AND (ABS a>99999999
 OR ABS a<.00001) THEN  BEEP .2,
10: BEEP .2,10: BEEP .2,10: BEEP
 .2,10: PRINT AT x,22;"********"
: GO TO 3000
3290 IF SGN a<>-1 THEN  PRINT AT
 x,17;">";AT x,22;a
3292 IF SGN a=-1 THEN  PRINT AT
x,17;">";AT x,21;a
3294 LET c$=STR$ a
3296 PRINT AT x,31;" ": LET x=x+
1: PRINT AT x,31; FLASH 1;"<"
3300 GO TO 3160
3330 LET c$=c$+b$
3340 GO TO 3040
3400 REM  ** subtotal **
3405 IF subtf=0 THEN  GO TO 3470
3410 LET a=VAL c$: LET c$=STR$ a
3420 IF a<>0 AND (ABS a>99999999
 OR ABS a<.00001) THEN  BEEP .2,
10: BEEP .2,10: BEEP .2,10: BEEP
 .2,10: PRINT AT x,22;"********"
: GO TO 3480
3430 PRINT AT x,17;">"
3440 IF SGN a<>-1 THEN  PRINT AT
 x,22;FN d(a)
3450 IF SGN a=-1 THEN  PRINT AT
x,21;FN d(a)
3460 PRINT AT x,31;" ": LET x=x+
1: IF x>=21 THEN  GO SUB newpage
3465 PRINT AT x,31; FLASH 1;"<"
3466 LET subtf=0: LET calcf=0
3470 RETURN
3500 PRINT AT x,16;"TOT": LET no
tec=0: LET cursf=0
3510 LET a=VAL c$: LET c$=""
3520 IF a<>0 AND (ABS a>99999999
 OR ABS a<.00001) THEN  BEEP .2,
10: BEEP .2,10: BEEP .2,10: BEEP
 .2,10: PRINT AT x,22;"********"
: GO TO 2000
3530 IF SGN a<>-1 THEN  PRINT AT
 x,22;FN d(a)
3540 IF SGN a=-1 THEN  PRINT AT
x,21;FN d(a)
3550 PRINT AT x,31;" ": LET x=x+
1: LET y=1: IF x>=21 THEN  GO SU
B newpage
3560 IF x=21 THEN  GO SUB newpag
e
3570 GO TO 2000
3600 REM ** delete/cursor **
3610 PRINT AT x,31; FLASH 1;"+"
3620 LET b$=INKEY$: IF b$=a$ THE
```

```
N  GO TO 3620
3630 LET a$=INKEY$: IF a$="" THE
N  GO TO 3630
3640 IF a$=CHR$ 9 THEN  PRINT AT
 x,31;" ": LET y=1: BEEP .1,25:
PRINT AT x,0; FLASH 1;"+": GO TO
 2520
3650 IF a$=CHR$ 10 THEN  GO TO 3
770
3660 IF a$=CHR$ 11 THEN  GO TO 3
700
3670 IF a$=CHR$ 12 THEN  GO TO 3
850
3680 LET b$=INKEY$: IF b$=a$ THE
N  GO TO 3680
3690 GO TO 3630
3700 REM ** cursor up **
3710 IF x=1 THEN  GO TO 3630
3720 PRINT AT x,31;" ";AT x-1,31
; FLASH 1;"+"
3730 LET x=x-1
3735 BEEP .05,25
3740 LET b$=INKEY$: IF b$=a$ THE
N  PAUSE 0
3750 GO TO 3630
3760 REM  ** cursor down **
3770 IF x=21 THEN  GO TO 3630
3780 PRINT AT x,31;" ";AT x+1,31
; FLASH 1;"+"
3790 LET x=x+1
3795 BEEP .05,25
3800 LET b$=INKEY$: IF b$=a$ THE
N  PAUSE 0
3810 GO TO 3630
3850 REM ** line delete **
3860 PRINT AT x,16;"
 "
3870 LET b$=INKEY$: IF b$=a$ THE
N  GO TO 3870
3880 BEEP .1,10
3890 GO TO 3630
4000 REM ** table **
4005 PRINT AT x,y;" "
4110 LET x=x+2: IF x>16 THEN  GO
 SUB newpage: LET x=2
4120 INPUT  FLASH 1;"Enter no. o
f columns req."; FLASH 0;" ";col
s: IF cols<2 OR cols>4 THEN  GO
TO 4120
4125 GO SUB colsok
4128 IF ans=no THEN  GO TO 4120
4130 DIM d$(cols,7): DIM e$(cols
,7)
4150 IF cols=2 THEN  PRINT AT x,
7;"(a)";AT x,22;"(d)": LET start
=5: LET space=8: GO TO 4180
4160 IF cols=3 THEN  PRINT AT x,
4;"(a)";AT x,14;"(b)";AT x,24;"(
d)": LET start=2: LET space=3: G
O TO 4180
4170 IF cols=4 THEN  PRINT AT x,
2;"(a)";AT x,10;"(b)";AT x,18;"(
c)";AT x,26;"(d)": LET start=0:
LET space=1: GO TO 4180
4180 LET x=x+1
4200 REM **col desc 1 **
4210 FOR e=1 TO cols
4220 LET z=start+(7+space)*(e-1)
4230 INPUT  FLASH 1;"Enter col.
descr."; FLASH 0;" "; LINE d$(e)
4240 PRINT AT x,z;d$(e)
4250 GO SUB info
4260 IF ans=no THEN  PRINT AT x,
z;"        ": GO TO 4230
4280 NEXT e
4300 REM ** formula **
4320 PRINT AT x-2,0;"d="
4330 INPUT  FLASH 1;"Enter formu
la"; FLASH 0;" d="; LINE b$: IF
LEN b$>30 THEN  GO TO 4330
4340 PRINT AT x-2,2;b$
4350 GO SUB info
4360 IF ans=no  THEN  PRINT AT x
-2,2;"
  ": GO TO 4330
4400 REM ** decimal places **
4410 GO SUB dpok
4420 IF ans=yes THEN  GO TO 4500
4450 INPUT  FLASH 1;"Enter dec.
places required"; FLASH 0;" ";dp
4460 IF dp>6 THEN  GO TO 4440
4470 LET dp=INT dp: GO TO 4410
4500 REM ** enter variables **
4505 LET totf=0: LET t=0: LET x=
x+2
4510 FOR e=1 TO cols-1
4520 INPUT  FLASH 1;("Enter ";d$
(e)); FLASH 0;" "; LINE a$
4530 IF LEN a$>7 THEN  BEEP .1,1
0: GO TO 4520
4540 LET e$(e)=a$
4550 GO SUB nocheck: IF nof=1 TH
EN  BEEP .1,20
4560 LET z=start+(7+space)*(e-1)
4570 PRINT AT x,z;e$(e)
4580 NEXT e
4590 GO SUB info
4600 IF ans=no  THEN  PRINT AT x
,0;f$: GO TO 4510
4601 REM ** calc total **
4602 LET aflag=0: LET bflag=0: L
ET cflag=0
4604 FOR e=1 TO LEN b$
4606 IF b$(e)="a" THEN  LET afla
g=1
4608 IF b$(e)="b" THEN  LET bfla
g=1
4610 IF b$(e)="c" THEN  LET cfla
```

```
g=1
4612 NEXT e
4615 IF cols>=2 AND aflag=1 THEN
  LET a=VAL e$(1)
4620 IF cols>=3 AND bflag=1 THEN
  LET b=VAL e$(2)
4630 IF cols=4 AND cflag=1 THEN
 LET c=VAL e$(3)
4640 LET d=VAL b$: LET t=t+d
4650 LET z=start+(7+space)*(cols
-1)
4660 IF d<>0 AND (ABS d>99999999
 OR ABS d<.00001) THEN  BEEP .2,
10: BEEP .2,10: BEEP .2,10: BEEP
 .2,10: PRINT AT x,z;"********":
 GO TO 4910
4665 IF cols=4 THEN  LET c$=STR$
 d: IF LEN c$>8 THEN  GO SUB 510
0: LET d=a
4670 IF SGN d<>-1 THEN  PRINT AT
 x,z;FN d(d)
4680 IF SGN d=-1 THEN  PRINT AT
x,z-1;FN d(d)
4920 GO SUB tableok
4930 IF ans=yes THEN  GO TO 5000
4960 LET x=x+1: IF x>=21 THEN  G
O SUB newpage: FOR e=1 TO cols:
LET z=start+(7+space)*(e-1): PRI
NT AT x,z;d$(e): NEXT e: LET x=x
+2
4970 GO TO 4510
5000 PRINT #1;TAB 1;"Total requi
red?"
5020 IF INKEY$="y" THEN  INPUT "
": LET x=x+1: GO TO 5045
5030 IF INKEY$="n" THEN  INPUT "
": LET t=0: GO TO 5070
5040 GO TO 5020
5045 IF t<>0 THEN  IF ABS t>9999
9999 OR ABS t<.00001 THEN  LET t
otf=1
5048 IF totf=0 AND cols=4 THEN
LET c$=STR$ t: IF LEN c$>8 THEN
 GO SUB 5100: LET t=a
5050 IF totf=0 AND SGN t<>-1 THE
N  PRINT AT x,z-7;"TOTAL ";AT x,
z;FN d(t)
5055 IF totf=0 AND SGN t=-1 THEN
  PRINT AT x,z-7;"TOTAL";AT x,z-
1;FN d(t)
5060 IF totf=1 THEN  PRINT AT x,
12;"TOTAL ";t
5070 LET totf=0: LET x=x+1
5080 IF x>=21 THEN  GO SUB newpa
ge
5090 GO TO mode
5100 FOR f=LEN c$ TO 1 STEP -1
5110 IF c$(f)="." THEN  GO TO 51
30
5120 NEXT f
5130 IF f<>8 THEN  LET a=(INT (1
0^(LEN c$-f-2)*VAL c$+.5))/10^(L
EN c$-f-2)
5135 IF f=8 THEN  LET t=INT (t+.
5)
5140 RETURN
9300 REM ** copy **
9310 COPY
9320 GO SUB copyok
9330 IF ans=no THEN  PRINT AT 0,
0: GO TO copy
9340 RETURN
9400 REM **  screen prompts **
9410 PRINT #1;"Copy OK?": GO TO
9500
9420 PRINT #1;"Information corre
ct?": GO TO 9500
9430 PRINT #1;TAB 1;"Screen full
.  Go to next page?": GO TO 9500
9440 PRINT #1;"Is table calc. co
mplete?": GO TO 9500
9450 PRINT #1;cols;" columns sel
ected -  OK?": GO TO 9500
9460 PRINT #1;"Maximum decimal p
laces ";dp;" - OK?": GO TO 9500
9500 IF INKEY$="n" THEN  LET ans
=0: INPUT "": GO TO 9530
9510 IF INKEY$="y" THEN  LET ans
=1: INPUT "": GO TO 9530
9520 GO TO 9500
9530 RETURN
9600 REM ** newpage **
9605 PRINT AT x,y;" "
9610 GO SUB copy
9620 LET x=1: LET y=1: LET page=
page+1
9630 INPUT AT 22,0;" ";"": PRINT
 AT 0,0;f$
9640 PRINT AT 0,24;"page ";page
9650 RETURN
9700 REM ** no check **
9702 LET dpcount=0: LET nof=0
9704 IF a$="" THEN  LET nof=1
9705 FOR f=1 TO LEN a$
9706 IF a$(f)="." THEN  LET dpco
unt=dpcount+1: IF dpcount>1 THEN
  LET nof=1
9720 IF a$(f)="-" AND (f>1 OR LE
N a$=1) THEN  LET nof=1
9730 IF a$(f)<>"j" AND a$(f)<>".
" AND (CODE a$(f)>57 OR CODE a$(
f)<48) THEN  LET nof=1
9740 NEXT f
9750 RETURN
9799 REM ** functions **
9800 DEF FN d(x)=(INT (10^dp*x+0
.5))/10^dp
9810 DEF FN p(x)=(f/100)*x
```

The Musical Midi

A specialist application for the Spectrum unveiled by Ray Elder.



Early one evening recently found me motoring east towards Basildon, I had been invited to the home of Mr. Mike Beacher for a demonstration of an interface and software he was producing to work with the latest development in musical keyboards.

The most expensive keyboard synthesiser instruments are now being fitted with what is known as a Midi interface, but in line with most electronic developments, the cost of these are rapidly decreasing. Casio plan to have one of the cheapest marketed before this is published at under £300.

So what is it and why was I going to see him rather than have one sent to me at the office?

Well to answer the last question first, when I rang Mike he was reluctant to send one as he insisted that to really appreciate the capabilities of the interface and software it needed to be seen in a professional studio environment. So off I went to visit him.

The Performer

The actual interface simply connects the instrument to the Spectrum. In fact Mike is producing interfaces for most computers, but said that he preferred the Spectrum because of it's large memory, cost and ease of use.

The interface in it's pre production state consisted of a small box connected to the Spectrum by a ribbon cable, and with three DIN sockets to the sound source. A pulsing LED indicated that the connection was OK.

What is special about this Midi system is that the signals to and from the interface are not musical notes or sound, but are pure electronic signals which contain all the information needed by the keyboard to produce the sound.

This means that not only the note, pitch and duration is sent but also the voice (type of sound), pressure, volume and all the other sophisticated information required to recreate a sound is being supplied.

It follows that if all this information can be stored (and what better than a computer to do so) and then sent back to the instrument, the sound can be recreated.

The computer actually operates and PLAYS the instrument!

And this is the exciting and innovative difference between the Midi interface and the usual sound output.

Program

Programs to control and make use of the possibilities of this system have been developed by Mike and a team of freelance programmers.

Mike is emminently suitable to work on this area as he is a very musically talented ex-music teacher, he is also full of enthusiasm and energy for this system.

Several programs are under development and several are available now. He demonstrated one which provided a library of sounds programmed for a synthesiser, although usually a selection of preset voices are provided, the full potential of the infinitely variable settings of the sound envelope is rarely explored. As well as providing an extensive built in set up library the program offered a visually stunning option to define your own sounds.

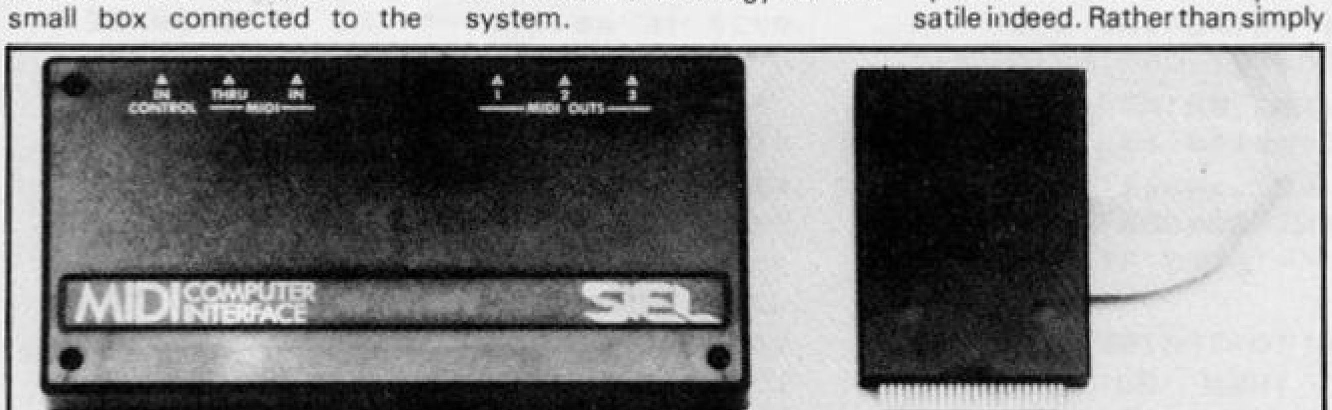
This was a simple to use way of approaching an extremely complex featured of modern keyboards, anyone who has been playing with one of the sound add ons will have some idea of the complexity of programming the comparatively simple AY chip. A cursor controlled visual display of all the parameters certainly helps when setting up your sound.

I could have played for hours with this program alone.

Another program allowed music which had been "recorded" to be displayed on both screen and printer in manuscript form, as this original music could be transcribed to any key under simple software control the possibilities for anyone involved in orchestral work were obvious.

The final program demonstrated was to me the most exciting and impressive of all. It was a program to turn the Spectrum into an eight channel recorder of the Midi signals.

Each tack has a wide range of options which make it very versatile indeed. Rather than simply list them I'll describe a demonstration Mike did which may give you some idea of the practical usage.



Performance

First he set his master synth. to a bass guitar voice, set the track to 1 and set a channel on the display to 2. He then "recorded" a piece of music.

Next Mike set his master synth (he had at least six instruments connected!) to a piano voice, set the channel on the display to 3 and recorded a ryhthm on track two, while listening to the original track being replayed.

He recorded five tracks in all each with a different channel number and then he set up each of the other synths.

Each synth could be switched to a channel corresponding to the channel number which was selected when recording and so on playback each synthesiser played back separately its own part in the appropriate voice. The sound was fantastic, as if a full group or orchestra were playing especially when the Midi drum machine was added with fully programmable and synchronised accompaniament.

Then of course there was so much control over speed and editing that even the best tape system couldn't match it.

Mike finally recorded the lot onto a cassette and I listened to it all the way back to the office while scheming how to persuade my boss that I ought to have one for the magazine! I also realised that Mike had been right to insist I visit him.

Encore

There are many more features and applications which are possible and I emphasise that this article is only scratching the surface of the capabilities of this system.

I forecast that soon no professional recording studio will be without a comparable system and that all serious musician will start to build a system of their own at home. Schools in particular would find it very valuable indeed, and some authorities and educational organisations have shown a good deal of interest.

Mike himself says that he finds it exciting that it will soon be within many people's budget to buy a keyboard, interface and computer and create high quality music which could be the basis of much more sophisticated systems as their equipment is added to and upgraded.

Tower of Hanoi

Kenneth Baker of Southampton has written the ultimate version of this game in machine Code for the 16K ZX81.

Invented about 100 years ago by the French mathematician Edouard Lucas, The Tower of Hanoi is perhaps the most fascinating and enduring of all puzzles. The object is to transfer the six parts of the Tower from position 'A' to position 'C' in the least possible moves, and without ever placing a larger block upon a smaller one. Position 'B' is used throughout as a temporary store.

To transfer a block from 'A' to 'B', simply input AB. If the move is valid the transfer will be made, and the score incremented by 1. The program is designed to reject any illegal moves.

In this six-tier version of the puzzle, the least possible moves is 63, and the maximum allowed is 99 — when the computer will decide that the player is doing so badly that it will automatically call a restart to the program. Restart can be called at any time by pressing 'R', and the only other active key, apart from the transfer keys ABC, is 'F', which will clear the program from memory.

To put the player on the right track, the first few moves are AB AC BC. . . but to say more might be to deprive the player of hours of hair-tearing frustration.

At the end of each game a caption containing a suitable comment as to the status of the final score will appear on the screen. Please remember that the least possible moves is 63.

The program is written entirely in Machine Code, which must be entered most carefully. The code is arranged in 139 lines of 11 HEX pairs: the first nine pairs representing the actual code, whilst the last two contain a checksum which is an addition of the code values in the line plus the line number. In this way, if any line should contain an error, or even if the wrong line is entered by mistake, it will be rejected with a request for the line to be re-entered.

Each line of 11 pairs should be entered together as a string, with one space between each pair. It should be noted, particularly if any other method of input is used, that the last two pairs are NOT instructions.

Making Space

The biggest problem with entering large Machine Code routines into the ZX81 is being able to create large enough REM statements in order to store them. The method chosen here is to form a REM statement of 120 characters in Line 4, and duplicating it 9 times in Lines 5-13 by the following method:

4 REM xxxxx(120 characters) xxxxx
EDIT
Change the Line No. to 5
Press N/L
EDIT
Change the Line No. to 6, and so on, until Line 13 is reached.

The size of Line 4 is then adjusted to encompass the 9 ensuing lines, which will result in a REM statement with the required number of 1254 bytes. When a REM statement is created in this way, it is most important to stabilize it with the addition of at least one line afterwards — hence the apparently redundant REM in Line 14 of the program, which must be left intact when the HEX LOADER is deleted.

Purely by the way, should you wish to use this technique in your own programs, the number of bytes per number of lines can be found by using the formula: $((C+6)/L)-6$ where C = the total characters required, and L = the number of lines. To avoid fractional answers, the odd byte or two can be added to the total characters. The best way to determine the number of lines is to factorize (C+6), again adding the odd byte until a convenient balance is reached. The length of the first line is then poked with (C+2).

```
   1 REM
   2 SLOW
   3 RAND USR 16731
   4 REM THIS LINE CONTAINS THE
         MACHINE CODE  READ TEXT
         BEFORE ENTERING THE
         PROGRAM
  14 REM
  15 POKE 16541,232
  16 POKE 16542,4
  17 LET Z=16544
  18 LET V=28
  19 FOR N=1 TO 139
  20 LET T=N
  21 LET C=1
  22 DIM A$(33)
  23 DIM A(9)
  24 PRINT AT 18,0;"INPUT LINE N
O. ";N
  25 INPUT A$
  26 FOR M=1 TO 25 STEP 3
  27 GOSUB 50
  28 LET A(C)=P
  29 LET T=T+P
  30 LET C=C+1
  31 NEXT M
  32 GOSUB 50
  33 LET Q=P
  34 LET M=M+3
  35 GOSUB 50
  36 LET P=256*Q+P
  37 IF NOT P=T THEN GOTO 39
  38 GOTO 43
  39 PRINT AT 18,0;"ERROR IN LIN
E ";N;AT 19,0;"PLEASE RE-ENTER"
  40 PAUSE 100
  41 CLS
  42 GOTO 20
  43 FOR M=1 TO 9
  44 POKE Z,A(M)
  45 LET Z=Z+1
  46 NEXT M
  47 NEXT N
  48 PRINT AT 18,0;"ALL CORRECT"
  49 GOTO 54
  50 LET I=CODE A$(M)-V
  51 LET F=CODE A$(M+1)-V
  52 LET P=16*I+F
  53 RETURN
  54 PAUSE 100
  55 CLEAR
```



When the code has been successfully entered, lines 15 to 55 can be removed, and the final instruction, before saving and running the program, is to POKE 16543,127: this will make the REM statement invisible, but can be omitted if my flair for cosmetic surgery is not to your taste.

```
001   00 7F 7F 7F 7F 7F 7F 7F 7F 03 F9
002   7F 7F 7F 7F 7F 7F 7F 7F 7F 04 79
003   7F 7F 7F 7F 7F 46 52 55 7F 03 EA
004   46 4B 43 55 48 7F 4B 54 7F 03 12
005   52 59 4C 4B 51 7F 7F 4E 59 03 3D
006   47 46 7F 4D 4B 44 55 7F 68 03 2A
007   68 7F 47 57 4B 48 55 7F 68 03 5B
008   68 7F 77 77 77 77 77 62 77 04 1B
009   77 77 77 77 7F 61 7F 77 77 04 32
010   77 77 7F 7F 60 7F 7F 77 77 04 42
011   77 7F 7F 7F 5F 7F 7F 7F 77 04 52
012   77 7F 7F 7F 7F 5E 7F 7F 7F 04 5A
013   7F 77 7F 7F 7F 7F 7F 5D 7F 04 5A
014   7F 7F 7F 7F 59 58 57 7F 48 03 D9
015   6B 48 55 47 46 59 48 46 7F 03 0A
016   54 6B 54 51 4C 51 47 52 7F 03 29
017   56 55 47 51 53 4C 7F 59 4C 03 17
018   56 7F 57 4B 4A 41 48 51 53 03 00
019   52 46 7F 58 41 7F 4F 55 4C 03 32
020   4C 55 46 52 7F 58 59 4F 55 03 21
021   48 7F 62 5A 5B 60 80 CD 79 04 19
022   41 C3 85 41 2A 22 40 36 00 02 A2
023   2A 0C 40 23 06 18 C5 06 20 01 B9
024   36 88 23 10 FB 23 C1 10 F4 03 EC
025   C9 21 A1 40 06 B9 7E 2F 77 03 C7
026   23 10 FA C9 CD 61 41 C3 C7 05 09
027   41 2A 0C 40 01 14 00 11 0E 01 06
028   01 19 EB 21 B6 40 ED B0 2A 03 FF
029   0C 40 01 19 00 11 6F 01 19 01 1D
030   EB 21 2F 41 ED B0 2A 0C 40 03 AD
031   01 14 00 11 B3 01 19 EB 21 02 1E
032   47 41 ED B0 21 34 40 36 FF 04 0F
033   3E C8 46 B8 20 FC C9 CD 8B 05 62
034   41 C3 0F 42 26 16 A1 40 21 02 B5
035   16 B5 40 21 16 A1 40 3F 0E 02 93
036   CA 40 11 0A D7 40 F6 0B E1 04 42
037   40 21 0B E8 40 21 0B F0 40 03 15
038   21 0B F9 40 21 0B 03 41 21 02 1C
039   0B 0E 41 47 01 19 41 09 01 01 2D
040   1A 41 09 01 1B 41 2D 0B 1C 01 3D
041   41 0F 0A 26 41 40 49 CD 61 02 A1
042   41 C3 4B 42 2A 0C 40 23 22 02 76
043   0D 42 21 CD 41 01 00 10 C5 02 7F
044   06 00 4E ED 5B 0D 42 EB 09 03 0B
045   EB ED 53 0D 42 23 06 00 4E 03 1E
046   23 D5 5E 23 56 23 22 49 42 02 CD
047   EB D1 ED B0 2A 49 42 C1 10 05 0E
048   DA C9 0D 42 CD F0 44 C3 79 05 5F
049   42 01 0A 0C CD F5 08 3E 89 03 1B
050   06 0B D7 10 FD 01 01 13 CD 03 09
051   F5 08 3E 89 06 0B D7 10 FD 03 EC
052   01 13 13 CD F5 08 3E 89 06 02 F2
053   0B D7 10 FD C9 CD 15 42 C3 04 D4
054   BD 42 AE 01 01 05 05 04 03 01 F7
055   02 01 00 96 01 07 00 00 00 00 D8
056   00 00 00 00 86 02 07 00 00 00 C7
057   00 00 00 00 00 00 AF 01 01 00 EA
058   06 05 04 03 02 01 00 97 01 00 E7
059   07 00 00 00 00 00 00 00 87 00 C9
060   02 07 00 00 00 00 00 00 00 00 45
061   00 CD 79 41 CD 51 42 CD 23 04 14
062   43 C9 00 00 00 00 CD BB 02 02 D4
063   EB 21 FD FD A7 ED 52 28 22 05 75
064   21 7F DF A7 ED 52 28 1D 21 04 0B
065   FE EF A7 ED 52 28 18 21 FD 05 72
066   EF A7 ED 52 28 13 21 FB EF 05 5D
067   A7 ED 52 28 0E 18 D2 3E 01 03 88
068   C9 3E 02 C9 3E 03 C9 3E 04 03 62
069   C9 3E 05 C9 00 00 00 00 00 02 1A
070   00 80 B0 AA BE 80 97 97 80 05 0C
071   88 88 88 88 88 88 88 88 88 05 0F
072   88 88 80 80 01 02 07 CD F5 04 24
073   08 11 0E 43 01 08 00 CD 6B 01 F4
074   0B CD CB 42 FE 04 28 3A FE 04 91
075   05 28 39 F5 C6 A5 F5 01 07 04 0E
076   07 CD F5 08 F1 06 01 D7 CD 04 B9
077   CB 42 FE 04 28 22 FE 05 28 03 D1
078   20 47 F1 F5 C5 B8 28 02 18 04 5A
079   03 C1 18 E9 F1 F5 C6 A5 F5 06 5A
080   01 08 07 CD F5 08 F1 06 01 03 22
081   D7 18 0A CD C3 03 C3 5B 41 04 3C
082   00 00 00 00 C1 F1 C3 88 43 03 92
083   00 00 00 00 00 00 FE 01 28 01 7A
084   08 FE 02 28 0D FE 03 28 12 02 CC
085   78 FE 02 28 16 FE 03 28 1A 03 4E
086   78 FE 01 28 1D FE 03 28 21 03 5C
087   78 FE 01 28 24 FE 02 28 28 03 6A
088   21 7F 42 11 89 42 18 26 21 02 75
089   7F 42 11 93 42 18 1E 21 89 02 E0
090   42 11 7F 42 18 16 21 89 42 02 88
091   11 93 42 18 0E 21 93 42 11 02 6E
092   7F 42 18 06 21 93 42 11 89 02 CB
093   42 C3 E6 43 76 00 05 76 00 03 7C
094   EA E5 D5 23 23 13 13 E5 D5 05 28
095   46 23 10 FD 7E EB 46 EB 13 04 82
096   10 FD EB 46 EB B8 28 02 30 04 9B
097   23 C1 C1 C1 C1 C1 C1 01 07 05 12
098   07 CD F5 08 01 02 00 11 21 02 68
099   43 CD 6B 0B 21 34 40 36 2D 02 E1
100   3E 0F 46 B8 20 FC C3 32 43 04 03
101   1B 7E EB 77 EB 36 00 E1 35 04 97
102   EB E1 34 E1 E5 4E 23 46 C5 05 A8
103   E1 CD FC 44 00 00 EB E1 73 05 94
104   23 72 EB ED 5B 0C 40 19 EB 04 80
105   E1 E5 E5 D5 4E 23 46 C5 E1 06 46
106   ED 5B 0C 40 19 D1 C3 60 44 04 4F
107   76 00 05 83 00 EA E5 01 0B 03 44
108   00 ED B0 00 11 16 43 E1 01 03 55
109   0B 00 EB ED B0 E1 4E 23 46 04 98
110   C5 E1 01 21 00 09 EB E1 73 04 7E
111   23 72 21 34 40 36 2D 3E 0F 02 49
112   46 B8 20 FC 21 B3 00 ED 5B 04 A6
113   0C 40 19 E5 21 EF 00 19 D1 03 B5
114   01 02 00 ED B0 01 07 07 CD 02 EE
115   F5 08 01 02 00 11 21 43 CD 02 B5
116   6B 0B 3A 9D 42 3C 32 9D 42 03 50
117   06 00 D6 0A FA BF 44 04 F2 04 4E
118   B6 44 C6 A6 F5 78 C6 9C 47 05 F2
119   F1 4F ED 5B 0C 40 21 C0 00 04 2C
120   19 70 23 71 C3 03 45 FE 01 03 9F
121   28 03 C3 32 43 CD 44 45 FE 04 30
122   04 28 09 FE 05 28 02 18 F3 02 E7
123   C3 5B 41 CD C3 03 11 7F 42 04 3F
124   21 9E 42 01 1F 00 ED B0 C9 04 03
125   01 21 00 A7 ED 42 C9 3A 9D 04 15
126   42 FE 64 D2 5B 41 3A 95 42 04 A1
127   C3 D6 44 00 80 37 2A 28 1B 03 80
128   00 1D 20 00 2B 2A 27 80 B3 02 6C
129   B4 B9 80 AC B4 B4 A9 80 80 06 2B
130   80 AB A6 AE B7 80 80 80 80 05 B8
131   AC B4 B4 A9 80 80 80 80 A7 05 E7
132   B7 AE B1 B1 AE A6 B3 B9 80 06 8B
133   3A 9D 42 FE 57 30 17 FE 4B 04 83
134   30 0D FE 40 30 05 21 39 45 02 D5
135   18 0D 21 2F 45 18 08 21 27 01 A9
136   45 18 03 21 1E 45 E5 01 12 02 64
137   07 CD F5 08 D1 01 0B 00 CD 04 04
138   6B 0B CD CB 42 C9 00 00 00 03 A3
139   00 00 00 00 00 00 76 00 00 01 01
```

Problem page



David Nowotnik answers your questions and offers advice on your programming techniques.

Saving ensured?

Dear David,
In a past issue of *ZX Computing* Peter Shaw suggested a method of ensuring the saving of programs on the ZX81 by connecting the save lead to the TV modulator. Is this a suitable solution for the same problem on the Spectrum?

Charles London,
Peckham,
London

Charles,
Such drastic action should never be necessary on the Spectrum. If you have any problems with load and save, try removing the save lead when loading and the load lead when saving. One reader has suggested placing a 330 ohm resistor in the load lead connection to overcome the need of constantly removing leads. Remember to use VERIFY every time you save a program to be certain all has gone well.

Calmer CLEARing

Dear David,
Some times I find that the only way I can completely clear a machine code program from the computer is to turn off then on again. Is there a gentler way I can achieve this?

Barrie Wiggins,
Friedrichsdorf,
West Germany.

Barrie,
If you get back to BASIC after the machine code is complete, then you can use the direct command RANDOMIZE USR 0 on the Spectrum (RAN USR 0 on the ZX81) to reset the computer and wipe out the machine code.

Tinkering with Timex

Dear David,
As a owner of the Timex 2068 I find few programs available apart from the 20 titles released by Timex. I would like to adapt programs available for the Spectrum for the 2068. Most BASIC programs work OK, but most machine language programs do not work. To help me, are the following available?
(1) An assembly language program for the 2068 computers.
(2) A translater to convert Spectrum machine code to 2068 machine code.
(3) A procedure to load Spectrum machine code programs into the 2068 for editing later.

Benedict Bader,
Ijamsville, USA

Benedict,
I'll answer your questions in the order given:
- *1) An assembler written entirely in BASIC for the Spectrum will work on the 2068. A good example of such a program appears in 'The Spectrum Pocket Book' by Trevor Toms (from Phipps Associates, 172 Kingston Road, Ewell, Surrey). The book also contains a good disassembler and a guide to the Spectrum ROM which should be of use to you.*
- *2) Both machines use Z80 machine language, so no translation is necessary; however I suspect what you want is a something to translate the Spectrum ROM to the 2068 ROM — that doesn't exist, I'm afraid.*
- *3) If you prepare a safe area of RAM by lowering RAMTOP (the CLEAR command), then load the machine code with LOAD "", CODE x, where x is an address above RAMTOP, then the machine code will be loaded to start at that address. You can then load a disassembler or editor program below RAMTOP to work on the machine code.*

Light Pen

Dear David,
I am looking for a light pen for my Spectrum. Do you know where I can obtain one?

Michelangelo Mari,
Latina,
Italy

Michel,
Dk'tronics (Unit 6, Shire Hill, Industrial Estate, Saffron Waldon, Essex, England) sell one for £19.95. It plugs into the earphone socket, and is complete with software.

Internal RAMpack

Dear David,
I have a 16K Spectrum and would like to get an internal RAMpack for it. Please could you advise where to send my Spectrum and how much it will cost.

Paul Menzies,
Glasgow.

Paul,
Several companies will sell you 'kits' to do the 32K RAM upgrade yourself, although most will do it for you for a slight additional cost. The important thing is to identify whether you have an issue 1 or issue 2/3 Spectrum. Take a look at the edge connector; on issue 1 Spectrums and spaces are slightly wider than the metallic strips. On issue 2 and issue 3 machines the strips are wider than the gaps between them. I got my trouble-free upgrade kit from Computer Add-ons (7-9 Thane Works, Thane Vills, London N7); for series 1 it cost £34.50 and for series 2/3 it is £20.50. Fitting is an extra £7.

Piracy with Privacy

Dear David,
Is there anyway I can save programs from proprietary cassettes e.g. "The Hobbit" on microdrive, or must I continue to use a tape recorder for loading these programs?

P Johnson,
Hamilton,
New Zealand

Mr. Johnson,
As software producers try hard to stop you copying their tapes to prevent piracy, so sales of programs which give you 'back-up' copies of commercial programs thrive. I don't want to enter the argument over who is in the right, but if you have a genuine need to transfer commercial software to microdrive cassettes solely for your own convenience and use, then I suggest you consult the back pages of the popular microcomputer magazines. I've noticed a couple of 'suppliers' who sell tape to microdrive copiers. But I offer no guarantees over their quality!

Note. This view is not necessarily shared by the editors or publishers of *ZX Computing*.

The Key to Success

Two keyboards reviewed: the old and the new.

This is an impressive looking piece of equipment, one of the largest keyboards I've seen.

Mancomp 184 Keyboard

The keys themselves have top quality switches with flat topped square plastic pads. At the moment the legends are stick on labels with larger than usual writing, very bright and easy to read, but the company tell me they are considering having them specially printed on the plastic.

A keypad is included on the right of the main board and a single entry full stop key has been added to it making numerical data entry easier. The main keys have a full sized space bar, double sized ENTER key, yellow caps shift and two extra of these keys, and extra cursor keys in bright green pairs at each side of the bottom of the keyboard. These keys have a very positive feel but tended to catch if not pressed centrally.

A hole in the top right of the case allows the power pack to be held inside the case and making for a much neater, more compact unit. The front of the case is angled and this makes it comfortable to use as it is just right for resting your wrists on when typing.

To install your Spectrum you remove the top of its case, place it in position in the new case, screw down two halves of the case together. An interface 1 can also be included in the case.

When tightening one of the retaining clips a small chip of fibreglass broke off and the screw would not grip and so the Spectrum base was left to wallow around inside the case.

Mancomp told me that it had found favour among teachers and so I tested it in a school where the pupils were very computer orientated and used to handling this kind of equipment.

Perhaps I had a faulty keyboard and maybe it was coincidence but on Tuesday it was installed, Thursday the keys started falling off and on Friday the Spectrum died. End of test, end of Spectrum, end of review.

The Ricoll Electronics Keyboard

I guess that phrases like 'up market' and 'down market' are still around. A certain sort of person used to label a certain kind of commodity this way. As so often with such expressions, they merely indicate a boringly obvious fact. Namely that given any sort of product, you will find examples which are on the one hand little more than functional. That is they do what is required of them without glamour or glitter. And on the other hand examples which have been given the luxury touch, or the executive touch, the super deluxe . . . and needless to say these are the most expensive. In between there is always something to suit everybody.

The keyboard which RICOLL ELECTRONICS have brought out for the Spectrum is a no nonsense steel housing which will put any office dweller in mind of filing cabinets and other office equipment with respect to appearances but not of course to size. Colour is similar too. Solid, sturdy and reliable, there are no trimmings, no smooth finish, indeed the finish is rough to the touch.

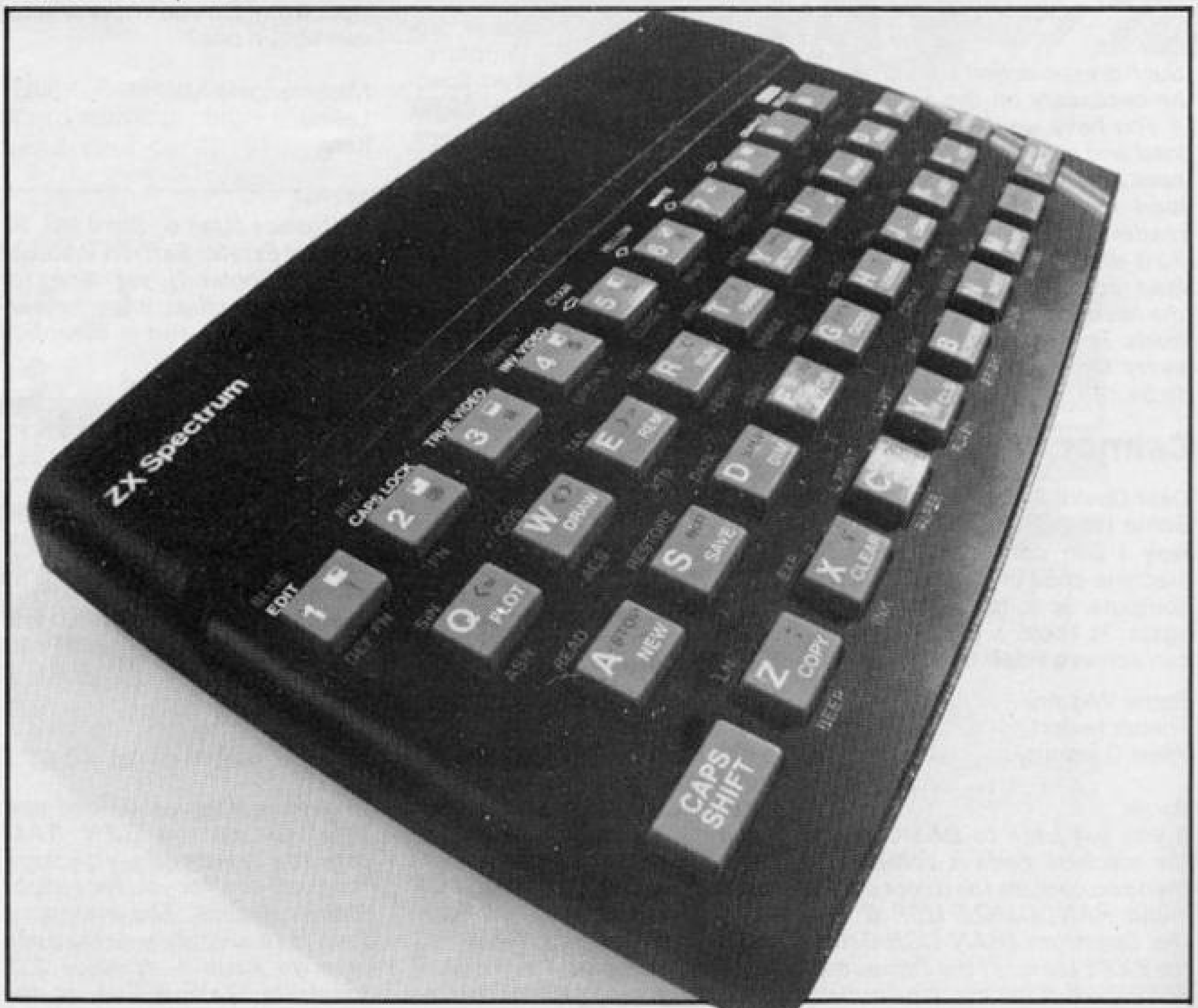
The space bar and keys follow the traditional typewriter pattern and transfering from the typewriter to this keyboard holds no problems. Nothing could be simpler or more straightforward. It's a keyboard. No more no less. I've hinted that this is a functional piece of equipment, the lettering and information on the keys in fact consists of transfers or stick on. They look vulnerable but do not come unstuck. They do collect marks around the edges however.

I've been using mine non stop now for several months and I can vouch for it's reliability and ease of use. Personally I find the keys positive and responsive and can use them for hours when writing without frustration or tiredness. When using it for programming the small size lettering will prove a problem to those not already familiar with the Spectrum keyboard.

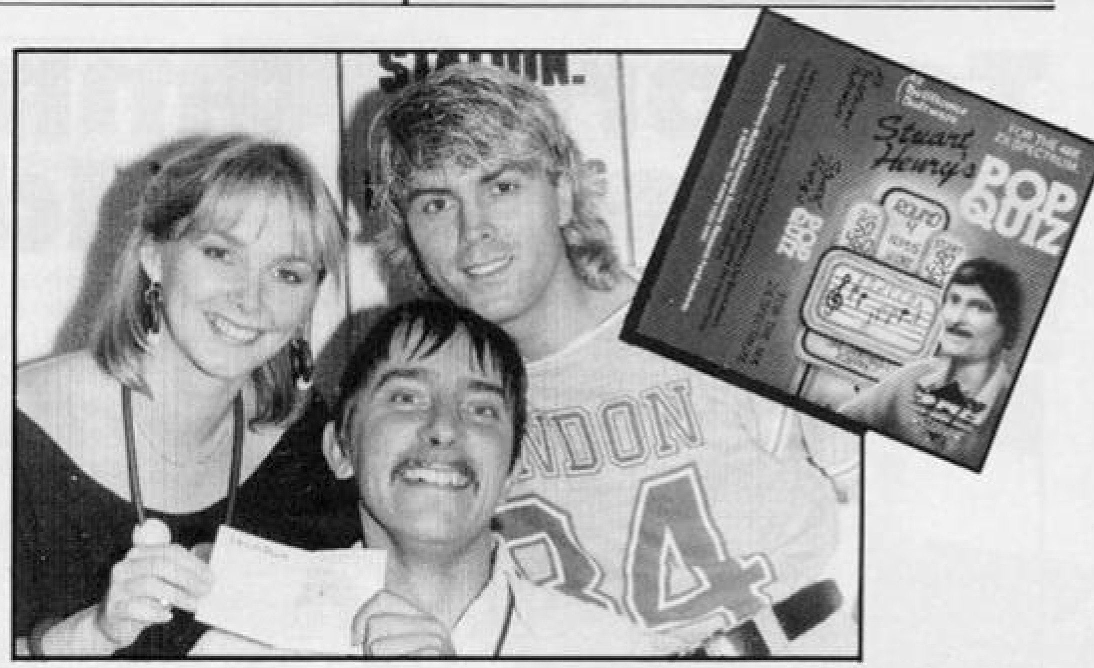
It's quite a heavy piece of equipment and obviously designed for heavy use. Well I've certainly given it that. And when I haven't been using it my sons have and they are no respectors of fragile technology. In short the keyboard has taken a fair old 'hammering' and still going strong.

It measures approximately 280mm × 210mm × 55mm and is mounted firmly on four solid rubber cushions. Installation is no problem provided the Spectrum is handled with the care and respect it deserves in it's exposed state while being transferred from one home to another. It stands to reason that disconnecting and reconnecting the ribbon cable PCBs, and fixing the board into the new housing must be done carefully. Even so it is not a task which requires much time, or expertise for the serious Spectrum User.

If looks are not important but solid hard wearing efficiency is, then many people will be more than satisfied with this keyboard.



Pop quiz helps multiple sclerosis research



Bucks Fizz stars Cheryl Baker and Mike Nolan joined the campaign to fight Multiple Sclerosis when they presented Radio Luxembourg disc jockey Stuart Henry with a cheque for £2,100 in London today (Wednesday).

The money has been raised by Bellflower Software and W.H. Smith & Son from sales of the computer game, "Stuart Henry's Pop Quiz", which has been challenging owners fo the 48K Sinclair Spectrum computer to show off their knowledge of pop music this summer.

Stuart Henry suffers from Multiple Sclerosis, a paralysing disease of the central nervous system, and the MS Society operates a research fund in his name. "Everyone who has bought the program so far has made a contribution to vital research into finding a cure for MS," said Bellflower director, David Gordon. "We hope more people will take up the challenge of 'Stuart Henry's Pop Quiz' so we can Have more presentations like today's."

Activision action

Activision, a company who found fame beating Atari at their own game, have joined the Spectrum set.

Their initial releases were Beamrider, Enduro, H.E.R.O., Pitfall II, River Raid, Space Shuttle, and Zenji and were all quite acceptable arcade games. Our reviewers will hopefully explain more about them. I had a quick preview of HERO and Enduro and really enjoyed the former, but was not too impressed ewot the latter.



The Designer's Pencil looks interesting from the press release we have been sent and appears to be a drawing and cound creation utility program.

By now they should have released their blockbuster, Ghostbuster! This is based on a film which has been a success in the USA (which means it will probably be a success over here), and the computer game will have been available before the film of Activision's plan work out.

Could it be a trend that we may soon see the film of the game! As a free service to all computer users, Activision are running a Software club. The aim is to give news, tips and views from other members, competitions, a worldwide penpal service and of course, an update of forthcoming Activsions programs.

More details from Activision Software club, 5/7 Forlease Rd., Maidenhead, Berks. SL6 1RP.

In . . . . Brief

● A superbly packaged and very atmospheric adventure game arrived from **Palace Software.**

Interestingly called "**Valkyrie 17**, The Red Kipper Flies At Midnight", it has a James Bond style theme and features both graphic and text locations.

Also included is a microdrive converter program, now there's a good idea.

● **Mirrorsoft** are going serious, their new range of programs are aimed at the home computer user who wishes to make "practical" use of his machine.

The titles in the series, called **Home Discovery**, are:

The Joffe Plan, a dieting program.

Know Your Own Personality, from the book of the same name.

Know Your Own Psi-Q, assesses your Psychic ability.

Mastermind and Quizmaster, puts you in the hot seat or in the questionmaster's chair.

Star Seeker and **Solar System**, for those interested in the night sky.

These are priced at £9.95 each and all have rather impressive names linked to them, could be worth consideration.

● Slightly less serious is "**Here and There with the Mr. Men**" which is aimed at teaching direction and planning to 4-8 year olds in a fun way.

When are we going to see my favourites "The Perishers" in a program I wonder?

● **Roybot**, 45 Hullbridge Rd. Rayleigh, Essex, SS6 9NL have produced an **Assembler & Machine Code Debugger**. It looks very impressive and some unusual features have been included, such as using the line numbers instead of labels (call L100 is recognised). It will cost you £9.95 or £13.95 for a microdrive version, and is meant as one of a series, **'Games designer Macro and routines"** and **"Teach yourself Machine code"** 'being two other titles in the pipeline.

● Brilliant **Micromega** have produced **Kentilla**, an adventure game, **Braxx Bluff**, one of the most difficult arcade adventure games I've ever tried to play, and Jasper, which will be a classic! The graphics in Jasper are superb it has a great theme and it's darn well addivtive.

Top programmer takes the gold



Top microcomputer writer Mike Hale has been presented with a gold cassette for researching sales of 100,000 cassettes, by software company, Hewson Consultants.

Sales of Mike's two programs, Nightflite II and Heathrow ATC topped the 100K mark at the end of September and Andrew Hewson, Managing Director of Hewson's felt the landmark deserved recognition.

Even more tangible evidence of Mike's programming success has come in the form of the Ralley Tampico single engine light aircraft Mike has bought with just some of the proceeds of his sales.

Mike is not only a keen microcomputer programmer of simulation games, but is also an air traffic controller at London's Heathrow airport and a keen amateur pilot. Both his job and his hobby have provided the inspiration for his two best selling games, which have that all important ingredent for simulations — realism.

Latest addition to Mike's prodigious output is a revised and enhanced version of Heathrow ATC for the 48K Spectrum, featuring a Concorde landing routine, enchanced graphics and help pages. On the B side of the cassette is Schipol ATC, a less demanding version of the simulation based on landing patterns at Amstersam's international airport.

Andrew Hewson commented, 'Mike's success is due to his imaginative ideas, programming skill and his ability to meet deadlines.'

More at less

A new name has been added to the list of companies marketing buget software.

Software Supersavers has produced a batch of programs for the 48K Spectrum at £2.99 each. They are a subsidiary of the well known Software Projects company who brought us Jet Set Willy.

Fred's Fan Factory, Ziggurat, Shuttle Shock, Moonlighter, Flip Flap, Loony Lander and Freex are the titles and I tried a few out.

All have a very uninformative cassette insert with just the operating keys explained, the rules and purpose of the game are not mentioned.

However, on screen a fairly comprehensive set of instructions were provided for those I looked at. **Flip Flap** is an interesting pinball game with twenty tables to be played. This is a very enjoyable game and would certainly have been sold for around £5.00 a few months back.

A few plays and the rough edges show, there is no information on using the "gamble" feature and holding down certain number keys causes the program to crash.

Perhaps we are too close to the trade though. I have known several 'gamesters' who have bought budget price software to be perfectly happy with it even though it has produced nothing but scorn from the critics.

As far as I can see, you pay your money and take your chance, and if you want enjoyable but uninspired games then these may be for you. All credit to Software Projects for not promoting them as full price software.

In . . . . Brief

- **Creative Sparks** are flying **Danger Mouse** in two games, **Double Trouble** and **Black Forest Chateau, Delta Wing** and **Black Hawk** — two dogfight type games, **Stagecoach** and, for some light relief, **Macbeth** by a well known author.

- **PSS** are getting spooky with the release of **Poltergeist**. We also look forward with interest to the eventual launch of "**Swords and Sorcery**" which they claim is the nearest thing to the Dungeons and Dragons role playing game that has been produced on a computer. I know quite a few fans waiting with money clutched in eager hands for its appearance in the shops.

- More low priced software from **Century City**, available from many shops the titles announced are **Laser Lord, Man Trap, Sentinels** and **Kamikaze Collector**.
None have arrived as yet for our perusal, but the cassette inserts look approriately gaudy.

- My congratulations to **Computerworld**, 208 Kent House Rd. Beckenham, Kent BR3 1JN for the funniest press release I've ever read.
The product is the "**World's First Computer Joke Book**" and is available from them for £4.95 (£8.95 on Microdrive).
The title says it all and I was most disappointed that a review copy was not sent.
PS. If Neil phones me I'll tell him the sequel to his insert joke!

- A little bird tells me that **British Telecom** has invested some of its profits in producing software. **Firebird** (it's about time someone did that to Buzby) has released a range of well produced and good value for money programs.



● You know those "tests" which appear in the papers every now and then which give your personality by answering questions like: "How often do your beat your cat? 1. Twice a week, 2. Five times a week, 3. One a day."

Well William Stuart Systems have a computer version for you to dabble with

Actually I should be serious, they have produced some very clever hardware (Voice recognition, Speech and sound units) backed up by some impressive software and so this may be a good program and of practical use. At £14.95 they are obviously serious.

Called **Life-Line** we would be very interested to review it in a sensible manner if we get a copy.

● **Hisoft** is a company that have a reputation for very high quality and serious programs. Their Pascal compiler is universally praised. After a long time two more programs have been added to their range.

"**C**" is a programming language for the Spectrum and one which I admit my ignorance. I promise to investigate and report back as soon as possible.

● "**Ultrakit**" is a versatile toolkit, a program that gives you extra commands to aid your programming (Renumber, Auto etc). and from the specifications looks like it deserves a closer examination.

"C" is £25.00 and "Ultrakit" is £9.45 from Hisoft, 180 High St. North, Dunstable, LU6 1AT.

● **Monitor Ltd.** PO Box 442, London NW7 2JF tell us that they now have version 2 of their **Microdrive Utility Program** on sale. The range of facilities is very comprehensive and includes routines to merge with Masterfile and Tasword II to enable them to run on the Microdrive.

They won't trust us with a review copy but offered us a demonstration. Unfortunately our schedule made this impractical.

● From **Semaphore**, CH-1283 La Plaine, (Geneve) Suisse, we had a letter describing their **Microdrive FIG Forth** program. It appears to be one of the first programs that takes advantage of the Microdrive's filing and access features.

This must be of interest to all those interested in Forth, and we look forward to being able to give a full report soon.

● **Mastertronic** are expanding their range, **Space Walk**, and **BMX Racers** have been released for the Spectrum and any day soon we could see classics such as Arcadia and Alchemist under their banner.

So far they have not seen fit to provide us with review copies, still, we live in hope!

● **Omega Software** is yet another budget price software house to emerge. This one is slightly different however, in that their programs are made up of some well known programs from established software houses, Anirog, CRL and Silversoft among them.

Titles for the Spectrum are **Copter, Handy Andy, Chamber of Horrors, Pool, Corruption** and **Galactic Patrol.** This latter is also available for the much neglected ZX81.

All these are priced at £1.99.

● At last! Not only the promised sequel to Sabre Wulf, but TWO sequels.

Underwurlde continues Sabreman's adventures in a different scenario and **Knight Lore** features 3D perspective. We'll rush the review copies to our drooling reviewer and report next issue, meanwhile check them out at your nearest shop, they are from **Ultimate** of course!

● Another temptation — sorry, prize — for players of Star Dreams' adventure game **The Sandman Cometh.** Up for grabs this time is a weekend for two in Paris for the lucky winner, computer pressmen excluded (curses foiled!).

The game is in two parts and was devised and written by the same person who wrote the Stranglers "Aural Quest" game. For hopeful holidaying adventure fans from **Star Dreams**, 17 Barn Close, Seaford, Sussex, for £10.95.

CCS continue to confuse!

CCS, Cases Computing Simulations, have been producing strategy games since April 1982 and they have consistently added to their range over the years.

In the last two years they have run a competition in conjunction with the Sinclair User magazine to find the best amateur strategic or adventure game.

These have formed the basis of some of their latest releases. "The Prince" is a very different adventure game in that up to four players can play interactively, and is also programmed largely in machine code. This was the winning program and was written by John Sherry of Keele, it will cost £7.95.





"1942 Mission" is another adventure game and was written by Thomas Frost from Angus, and "Insurgency" and "War Zone" are war games, Insurgency for two players. These were written by Nicholas Holgate and Steven Thomas respectively.

The final program of this group is "Blue Riband" which is a seafaring navigational simulation, and was written by David Bark of Stranraer. All four of these are priced at £5.95 and all of these are for the 48K Spectrum.

CCS have a great deal of experience in marketing these type of games and if this is where your interest lies then you should check them out as soon as possible.

And now for something completely different....

As far as I know the release of "Nuke Lear" and "Tomb of Akhenaten" on their Charlie Charlie Sugar series of budget programs marks the first ever venture into arcade style games by this company. Both look very interesting and I'll get our tame experts to check them out. These are priced at £2.99 each and will soon be joined by Wizard's Curse, Voffy, Shift Shaft and 180.

Then they return to the style for which they're known and loved with Air Defence and The Staff of Zaranol, a wargame and an adventure game respectively. These cost £5.95.

And finally (for now!) such is the output of this company that just as I was finishing this outline, yet another two programs from them dropped through the letter box.

Barrowquest is another adventure by John Sherry and Superpower is a strategy game.

This should keep Greg Turnbull burning the midnight oil!

A great selection of challenging games.

Patience IV: The Wall

Colin Gooch continues his acclaimed series of card games for the Spectrum.

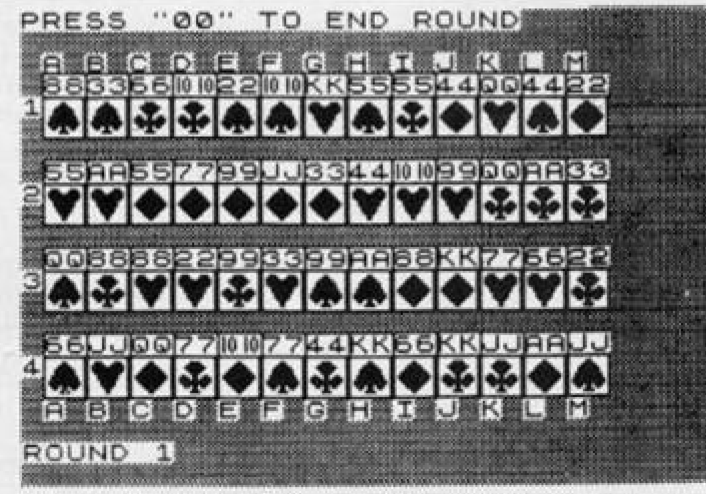
SPECTRUM PROGRAM

This is, I think, one of the most challenging of the wide range of patience games. Though obviously the run of cards plays an important part the moves that the player makes also affect the more the game. The full rules are in the programme, but briefly we lay out all 52 cards, remove the aces and then use the gaps to manipulate the cards into order.

From the programming point of view producing 52 cards on screen is quite a challenge. There is no way that a 13 x 4 grid of simulated cards is possible as we are restricted to a card width of two print positions. What we have as a "card" is a four by four print position representation of a diamond spade etc. underneath a value. The picture of each suit is made up if four user defined graphics.

Each card is contained in a pack and is represented by five elements a name, two to represent value, the suit and the colour. The programme reads the various elements to print out each card.

The pack remains as A$ and five blanks are inserted each time a card is removed and five blanks replaced at the new position. At the end of a round wrongly placed cards are transferred to another string P$, complete with gaps. This is shuffled, and the gaps replaced by aces. This is then printed out, a new card going to each location where there is no existing card, sensed by the ATTR function, which then becomes the new A$.



WHICH CARD TO MOVE? ENTER LINE THEN COLUMN

Line to line

1000	Defined functions to fix position of card on screen and in the various strings.
1010	Sets colours and CAPS SHIFT.
1020	Calls instructions and sets up pack.
1030-1070	Prints hand at the start of each round. Each is referenced by row and column.
1090-1140	The aces are removed from the display.
1150-1280	This forms a main loop which is repeated throughout a round: it invites you to enter a card to be moved. It then checks through each gap held in G$ to see if one is an appropriate move. A special set of conditions is checked if it is a two which is moved to the start of a line and thus does not follow another card. A faulty input will be ignored and you must enter again.
1300-1380	Subroutine to print a card.
1390-1400	Subroutine to blank a card.
1400-1440	Determines the position of card to be printed.
1450-1590	Programme is directed to this point whenever you resign at the end of a round. If round number is three or less the pack will be reshuffled and dealt again. If round is over three then you will be given an opportunity either to carry on or end the game.
1600-1680	If game is to end the programmer is directed to this section.
1690-1780	Subroutine to check that an input is the correct length and within the required limits.
1790-1880	The pack is set up and shuffled ready for play.
1900	The graphics are set up from DATA at end of the programme.
1920-2060	Instructions for play. Option to read.
2080-2250	Graphics DATA.
9980-9993	Start save and load routines. It is important to auto start at 9980 or graphics will not form.

Variables used

A$	assembled pack
B$	suits
C$	card in play
D$	temporary string
E$	holds four aces
G$	card immediately before blank
I$	title
N$ R$	Captions
P$	pack of incorrect cards
Q$	temporary string
Z$	input
ACE	slicer for E$
B,M,N,P,Q	loops
C	counting variable
COL	colour of card
CREP	location of card to be replaced into pack
F	temporary flag
RES	flag to determine end of game
ROUND	round in play
X,Y	position of card
z1,z2	values derived from Z$
Z1B,Z2B	temporary storage of z1,z2

Note

The amount and size of strings to be manipulated mean that the program will not run in 16K RAM.

```
   1 REM ************************
         *Underlined characters*
         *are entered in        *
         *GRAPHICS mode.        *
         ************************
1000 DEF FN Y()=((Z1-1)*13+Z2)*5
 -4: DEF FN Y$()=A$(FN Y() TO FN
Y()+4): DEF FN W$()=A$(FN Y()-5
TO FN Y()-1)
1010 POKE 23658,8: BORDER 4: PAP
ER 4: INK 0: CLS
1020 GO SUB 1910: INK 0: GO SUB
```

```
1790
1030 REM PRINT HAND
1040 FOR X=1 TO 25 STEP 2: FOR N
=2 TO 18 STEP 16: PRINT AT N,X;
PAPER 6;CHR$ (INT X/2+64): NEXT
N: NEXT X
1050 FOR Y=4 TO 18 STEP 4: PRINT
 AT Y,0; PAPER 6;Y/4: NEXT Y
1060 LET ROUND=ROUND+1: PRINT AT
 0,0; PAPER 6;"PRESS ""00"" TO E
ND ROUND";AT 20,0;"ROUND ";ROUND
: LET C=1: FOR Y=3 TO 18 STEP 4:
 FOR X=1 TO 25 STEP 2: IF ATTR (
Y+1,X)=32 THEN LET C$=A$(C TO C
+4): GO SUB 1290
1070 LET C=C+5: NEXT X: NEXT Y:
1080 REM REMOVE ACES
1090 PRINT #1;AT 0,0;"YOU MUST R
EMOVE THE ACES. ";AT 1,0;"PLEASE
 PRESS ENTER TO CONTINUE": PAUSE
 0
1100 LET G$=""
1110 FOR B=1 TO 4
1120 PRINT AT 21,0;"ENTER COORDI
NATES OF ACE No. ";B: INPUT LIN
E Z$: GO SUB 1690: GO SUB 1410:
IF A$(FN Y())<>"A" THEN PRINT #
1;"SORRY NOT AN ACE ": PAUSE 100
: INPUT "": GO TO 1120
1130 GO SUB 1400: LET G$=G$+Z$:
LET A$(FN Y() TO FN Y()+4)="
 ": NEXT B
1140 PRINT AT 21,0; PAPER 4;"
                             "
1150 REM MOVE A CARD
1160 INPUT AT 0,0;"WHICH CARD TO
 MOVE? ENTER LINE";AT 1,0;"THEN
COLUMN EG.3F ": LINE Z$
1170 IF Z$="00" THEN GO TO 1450
1180 GO SUB 1690: LET O$=Z$: GO
SUB 1410: LET C$=FN X$()
1190 IF C$="     " THEN INPUT ;
 AT 0,0;"SORRY BUT THAT WAS A SPA
CE!";AT 1,0;"PLEASE ENTER AGAIN"
; LINE Z$: GO TO 1170
1200 LET A$(FN Y() TO FN Y()+4)=
"     ": LET Z1B=Z1: LET Z2B=Z2:
 GO SUB 1400
1210 IF C$(1)="2" THEN FOR N=1
TO 8 STEP 2: IF G$(N+1)="A" THEN
  LET Z$=G$(N TO N+1): GO SUB 16
90: GO SUB 1410: LET NN=N: GO TO
 1270
1220 IF C$(1)="2" THEN NEXT N:
GO TO 1260
1230 FOR N=1 TO 8 STEP 2: LET Z$
=G$(N TO N+1): GO SUB 1690: GO S
UB 1410: LET H$=FN W$(): IF H$="
     " THEN NEXT N: IF H$="
 " THEN GO TO 1260
1240 IF VAL C$(2 TO 3)=VAL H$(2
TO 3)+1 AND C$(4)=H$(4) THEN LE
T NN=N: GO TO 1270
1250 NEXT N
1260 LET Z1=Z1B: LET Z2=Z2B: GO
SUB 1410: LET A$(FN Y() TO FN Y(
)+4)=C$: LET Y=Y+1: GO SUB 1290:
 GO TO 1150
1270 LET Y=Y+1: LET A$(FN Y() TO
 FN Y()+4)=C$: GO SUB 1290: LET
G$(NN TO NN+1)=O$
1280 GO TO 1150
1290 REM PRINT A CARD
1300 INK VAL C$(5): PAPER 7
1310 IF C$(1)="T" THEN PRINT AT
 Y,X;"QQ"
1320 IF C$(1)<>"T" THEN PRINT A
T Y,X;C$(1);C$(1)
1330 PLOT X*8,168-(Y*8): INK 0:
DRAW 0,8: DRAW 16,0: INK VAL C$(
5)
1340 IF C$(4)="S" THEN PRINT AT
 Y+1,X;"AB";AT Y+2,X;"CD"
1350 IF C$(4)="D" THEN PRINT AT
 Y+1,X;"EF";AT Y+2,X;"GH"
1360 IF C$(4)="C" THEN PRINT AT
 Y+1,X;"IJ";AT Y+2,X;"KL"
1370 IF C$(4)="H" THEN PRINT AT
 Y+1,X;"MN";AT Y+2,X;"OP"
1380 RETURN
1390 REM BLANK A CARD
1400 FOR N=1 TO 3: PRINT AT Y+N,
X; PAPER 4; INK 0;"  ": NEXT N:
RETURN
1410 REM CARD COORDINATES
1420 LET X=Z2*2-1
1430 LET Y=(Z1*4)-2
1440 RETURN
1450 REM END A ROUND
1460 LET ACE=1: LET P$="": LET F
=0: FOR M=1 TO 260 STEP 5: IF A$
(M)="2" AND (M=1 OR M=66 OR M=13
1 OR M=196) THEN LET F=1: NEXT
M
1470 IF A$(M TO M+4)="     " THE
N LET F=0: LET P$=P$+E$(ACE TO
 ACE+4): LET ACE=ACE+5: NEXT M
1480 IF P$=E$ THEN LET RES=0: G
O TO 1600
1490 IF NOT F THEN GO TO 1520
1500 IF VAL A$(M+1 TO M+2)=VAL A
$(M-4 TO M-3)+1 AND A$(M+3)=A$(M
-2) THEN NEXT M
1510 LET F=0
1520 IF M<=260 THEN LET X=(M/5-
INT (M/65)*13)*2+1: LET Y=INT (M
```

```
/65+1)*4 2: GO SUB 1390: LET P$=
P$+A$(M TO M+4): LET A$(M TO M+4
)="     ": NEXT M: IF P$="" THEN
 LET RES=0: GO TO 1600
1530 IF ROUND<3 THEN  GO TO 1560
1540 PRINT AT 19,0; PAPER 7; INK
 0;"YOU ARE ONLY SUPPOSED TO HAV
E  ";AT 20,0;"THREE GOES IF YOU
WANT TO CHEAT";AT 21,0;"AND KEEP
 GOING ENTER C. IF NOT ": GO SUB
 1770
1550 IF Z$="" THEN  LET RES=1: G
O TO 1600
1560 GO SUB 1760: PRINT AT 21,0;
">>>>>>>>>S H U F F L I N G <<<<<
<": FOR M=1 TO 4: GO SUB 1850
1570 GO SUB 1760
1580 LET CREP=1: FOR N=1 TO 260
STEP 5: IF A$(N)=" " THEN  LET A
$(N TO N+4)=P$(CREP TO CREP+4):
LET CREP=CREP+5
1590 NEXT N: GO TO 1060
1600 REM END GAME
1610 BORDER 1: PAPER 6: INK 0: C
LS
1620 IF RES THEN  FOR N=30 TO 1
STEP -1: BEEP N/100,N: PRINT AT
1,31-N;"\": NEXT N
1630 IF RES THEN  PRINT AT 3,3;"
SORRY BUT YOU DIDN'T MAKE  ";AT
4,1;"IT THAT TIME": IF ROUND>3 T
HEN  PRINT AT 4,13;"..EVEN THOUG
H YOU";AT 5,10;"HAD ";ROUND;" RO
UNDS"
1640 IF NOT RES  THEN  FOR N=1 T
O 30: FOR M=1 TO 6 STEP 5: BEEP
((31-N)/100)*(M=1),N: PRINT AT M
,N; FLASH 1; PAPER 7; BRIGHT 0;
INK 2;"!": NEXT M: NEXT N
1650 IF NOT RES THEN  LET N$="GR
EAT! YOU DID IT": FOR N=1 TO 17:
 PRINT AT 3,N+5; PAPER 3; INK 7;
 BRIGHT 1;N$(N): BEEP .01,N: NEX
T N: IF ROUND>3 THEN  PRINT AT 4
,5;"BUT TOOK  ";ROUND-3;" EXTRA
ROUND": IF ROUND>4 THEN  PRINT A
T 4,28;"S"
1660 PRINT AT 12,3;"YOU MAY:-";A
T 14,3;"1) PLAY THIS GAME AGAIN"
;AT 15,3;"2)PLAY NEXT GAME ON TA
PE";AT 17,5;"PRESS KEY TO CHOOSE
"
1670 PAUSE 0: IF INKEY$="1" THEN
 PAPER 4: INK 0: BORDER 4: CLS
: CLEAR : GO SUB 1790: GO TO 103
0
1680 IF INKEY$="2" THEN  CLS : P
RINT AT 10,10; FLASH 1;"START TH
E TAPE": LOAD ""
1690 REM CHECK VALID INPUT
1700 IF LEN Z$<>2 THEN  GO TO 17
50
1710 FOR V=1 TO 2: IF Z$(V)="" T
HEN  GO TO 1750
1720 NEXT V
1730 IF CODE Z$(1)<49 OR CODE Z$
(1)>52 OR CODE Z$(2)<65 OR CODE
Z$(2)>77 THEN  GO TO 1750
1740 LET Z1=VAL Z$(1): LET Z2=CO
DE Z$(2)-64: RETURN
1750 INPUT AT 0,0;"ENTRY UNACCEP
TABLE";AT 1,0;"PLEASE CHOOSE AGA
IN"; LINE Z$: GO TO 1690
1760 FOR N=19 TO 21: PRINT AT N,
0; PAPER 4;"
          ": NEXT N: RETURN
1770 INPUT AT 0,0; INK 0; PAPER
7;"PRESS ENTER TO CONTINUE
  "; LINE Z$: RETURN
1780 REM SET UP PACK
1790 PRINT AT 3,6;"PREPARING THE
 PACK": LET ROUND=0: LET P$="":
LET B$="SDCH": LET D$="A01202303
404505606707808909T10J11Q12K13":
 LET E$="A01S0A01C0A01H2A01D2"
1800 FOR Q=1 TO 39 STEP 3: FOR P
=1 TO 4: LET COL=0: IF P=2 OR P=
4 THEN  LET COL=2
1810 LET P$=P$+D$(Q TO Q+2)+B$(P
)+STR$ COL
1820 NEXT P: NEXT Q
1830 CLS : PRINT AT 7,9; PAPER 5
; INK 1;"ABEFIJMNIJEFAB";AT 8,9;
"CDGHKLOPKLGHCD";AT 10,12; PAPER
 7; FLASH 1;"SHUFFLING";AT 12,9;
 PAPER 5; INK 1; FLASH 0;"ABEFIJ
MNIJEFAB";AT 13,9;"CDGHKLOPKLGHC
D"
1840 FOR M=1 TO 4: LET A=21+(5*(
1+INT (RND*20))): LET P$=P$(A TO
 )+P$( TO A-1)
1850 FOR N=6 TO LEN P$-5 STEP 5*
(1+INT (RND*4)): LET C$=P$(N TO
N+4): LET P$=P$( TO N-1)+P$(N+5
TO )+C$: BEEP .01,N/10-10
1860 NEXT N: NEXT M: IF ROUND>0
THEN  RETURN
1870 NEXT N: NEXT M: CLS : LET A
$=P$: RETURN
1880 INPUT "PRESS ENTER TO CONTI
NUE"; LINE Z$: BEEP .05,10: RETU
RN
1890 REM GRAHICS
1900 PRINT AT 2,8;"FORMING GRAPH
ICS": RESTORE 2070: FOR N=1 TO 1
8: READ R$: FOR M=0 TO 7: BORDER
```

```
 M: READ R: POKE USR R$+M,R: NEX
T M: NEXT N: RETURN
1910 REM INSTRUCTIONS
1920 LET I$=" *** * * ***  * * *
  *  *   *      *  * * *    * * *
 * * *   *      *  *** **   * * *
 *** *   *      *  * * *    * * *
 * * *   *      *  * * ***   * *
 * * *** *** "
1930 PAPER 4: CLS
1940 INK 2: PRINT AT 4,Ø; PAPER
4; INK 4;I$;AT 12,6; PAPER 6; IN
K Ø;"A GAME OF PATIENCE";AT 16,2
Ø;"by C.N.GOOCH";AT 21,Ø;"  PRE
SS ANY KEY TO CONTINUE   ": PAUS
E 2Ø: INK 4: PRINT AT 4,Ø; INK 1
;I$: PAUSE 2Ø: IF INKEY$<>"" THE
N  GO TO 196Ø
195Ø GO TO 194Ø
196Ø IF INKEY$<>"" THEN  GO TO 1
96Ø
197Ø PRINT AT 2Ø,Ø; INK Ø; PAPER
 5;"    DO YOU WANT INSTRUCTIONS
?    ";AT 21,Ø;"     PRESS ""Y"
"ES OR  ""N""O      "
198Ø IF INKEY$="N" THEN  CLS : R
ETURN
199Ø IF INKEY$="Y" THEN  GO TO 2
Ø1Ø
2ØØØ GO TO 197Ø
2Ø1Ø INK Ø: CLS : PRINT '" A PAC
K OF CARDS WILL BE DEALT "'" INT
O FOUR ROWS OF THIRTEEN  "'" CAR
DS."'" THE ROWS DOWN THE SCREEN
ARE"'" NUMBERED 1 TO 4;AND THE "
'" COLUMNS ACROSS ARE A TO M.   "
'" CARDS ARE REFERENCED BY ROW "
'" FOLLOWED BY COLUMN EG.3K "
2Ø2Ø PRINT '" YOU MUST FIRST OF
ALL ENTER "'" THE REFERENCES OF
THE ACES    "'" THESE WILL BE REMO
VED."
2Ø3Ø PRINT '" THE OBJECT IS TO G
ET FOUR    "'" ROWS RUNNING FROM
2 TO  KING"'" THIS IS DONE BY M
OVING CARDS"'" INTO GAPS SUCH TH
AT THE CARD"'" TO THE LEFT IS TH
E ONE BEFORE"'" IT IN ACE TO KIN
G SEQUENCE, "'" IN EACH SUIT": G
O SUB 177Ø
2Ø4Ø CLS : PRINT '" YOU WILL NOT
 BE ALLOWED TO "'" MAKE AN ILLEG
AL MOVE. IN THE"'" EVENT OF AN I
NCORRECT ENTRY"'" YOU WILL BE AS
KED TO ENTER"'" AGAIN. THE CARD
CONCERNED "'" WILL FLICKER ONCE
TO SHOW "'" THAT AN INPUT WAS RE
CEIVED."
2Ø5Ø PRINT '" UNLIKE MANY GAMES
OF PATIENCE"'" THIS GAME DOES NO
T RELY PURELY"'" ON CHANCE. YOUR
 CHOICE OF CARD"'" TO BE MOVED W
ILL ALTER THE "'" PROGRESS OF TH
E GAME."
2Ø6Ø  PRINT '" THE GAME CONSISTS
 OF THREE "'" ROUNDS. WHEN YOU C
AN MOVE "'" NO MORE PRESS ""ØØ""
": GO SUB 177Ø: CLS : RETURN
2Ø7Ø REM GRAPHICS DATA
2Ø8Ø DATA "A",255,128,129,131,13
5,143,143,159
2Ø9Ø DATA "B",255,1,129,193,225,
241,241,249
21ØØ DATA "C",159,191,191,157,13
7,131,128,255
211Ø DATA "D",249,253,253,185,14
5,193,1,255
212Ø DATA "E",255,128,129,131,13
5,143,159,191
213Ø DATA "F",255,1,129,193,225,
241,249,253
214Ø DATA "G",191,159,143,135,13
1,129,128,255
215Ø DATA "H",253,249,241,225,19
3,129,1,255
216Ø DATA "I",255,128,131,135,14
3,135,131,153
217Ø DATA "J",255,1,193,225,241,
225,193,153
218Ø DATA "K",189,191,159,141,12
9,131,128,255
219Ø DATA "L",189,253,249,177,12
9,193,1,255
22ØØ DATA "M",255,128,156,19Ø,19
1,191,159,143
221Ø DATA "N",255,1,57,125,253,2
53,249,241
222Ø DATA "O",143,135,135,131,12
9,129,128,255
223Ø DATA "P",241,225,225,193,12
9,129,1,255
224Ø DATA "Q",Ø,36,42,42,42,42,3
6,Ø
225Ø DATA "R",Ø,72,84,84,84,84,7
2,Ø
9ØØØ RETURN
998Ø CLS : PRINT AT 1Ø,1Ø; FLASH
 1;" STOP THE TAPE ": FOR N=1 TO
 1Ø: BEEP .1,N: PAUSE 1Ø: NEXT N
: GO SUB 19ØØ: GO TO 1ØØØ
999Ø SAVE "patienceD" LINE 998Ø
9992 CLS : PRINT "      REWIND TH
E TAPE"'"        PLAYBACK TO VERI
FY"
9993 VERIFY "patienceD": CLS : P
RINT ; FLASH 1;"  ALL  OK"
```

Chess Clock

Use your ZX81 as an impartial time keeper when playing that next game of chess! The Rev. J H Wall of Newcastle shows you how.

Fed up with thrashing "Grandmaster Chess" at level ten? Why not take on a real live opponent — your son for instance (*or your old man — Ed.*) You might show him a thing or two with the old chess pieces even if you can't get near his score on Galaxians! Here's a program to help you — help you play, that is not win! It's a fully automatic chess clock.

I was driven to write CHESSCLOCK for our poor old dusty ZX81 because I couldn't get near the Spectrum for all the burntout space ships. This program does all that a proper chess clock would do, and much more besides. That could save you £20 if you were buying one for home use.

Tournament clocks

The chess clock used in tournaments is just two cheap clock meovements joined together. A lever with two buttons allows you to start one clock when you stop the other. Two little flags show when you have run out of time.

This simple program copies the chess clock, but in digital form. In addition, the program shows you how many moves you still must make before your time runs out. When you have made the required number of moves within the time allowed, the program automatically adds your next ration of time and the extra moves you have to make. Should you exceed the time limit, the display informs you of the bad news ie you have lost on time.

Program description

Lines 10-100 ask you to set the move-rate, and the duration of the first and subsequent sessions. Lines 110 to 292 set up the initial values of the variables and create the display. The 300's are the routines for updating the move number and the *actual* clock for White. Black is catered for in the same way in the 400's. 500 and 600 contain the bad news if you lose on time. The routine at 800 updates the variables when you have made all the moves required for one session. Lines 900 to 920 enables you to stop both clocks while you have tea or write down a sealed move.

The main problem in writing the program was to make both digital clocks as accurate as possible. White's clock is contained within lines 350-398, with Black's almost identical routine in 450 to 498.

The use of PAUSE though obvious is far too crude and actually doesn't allow for sufficiently accurate adjustment. Satisfactory adjustment was achieved by the FOR-NEXT loop at lines 372-376 (black 472-476) for basic setting, and by a couple of dodges for fine tuning. Lines 371 and 471 have extra spaces on the end of "STOP" which fractionally slow the routine, and for Black's clock, line 474 — which apparently prints nothing — gives a repeated adjustment in the loop. Incidentally, PRINT AT 0,0 takes far more time than PRINT AT 21,12 though I have no idea why! I would be very interested to know if every ZX81 will work with the same adjustments or whether you will need to make provisions for your own machine. I achieved an accuracy of 1/5th of a second in 5 minutes for both clocks.

How to use CHESSCLOCK

Set up your board with the ZX81, with the VDU on White's left. Choose a move-rate. 16 moves per hour is Grandmaster stuff, 24 per hour is usual Club rate and novices probably play at around 60 moves per hour!

Next. decide how long your playing session will be to the first time-check. (Remember this is the time for each player so the session could be twice as long).

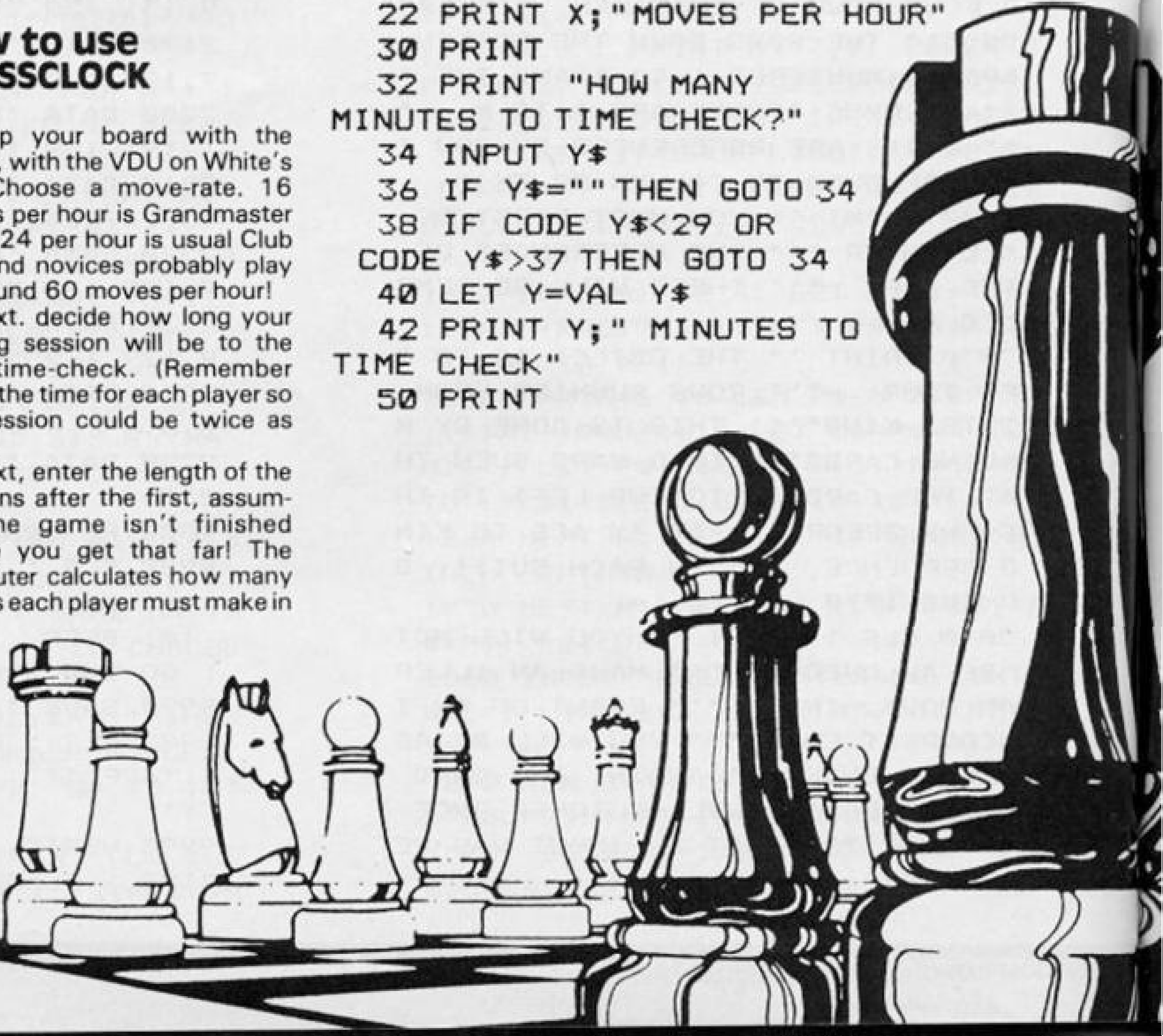
Next, enter the length of the sessions after the first, assuming the game isn't finished before you get that far! The computer calculates how many moves each player must make in the first session and then invites you to begin.

The display shows you how many moves are stil to be made and the time left for each player. When White has made his first move he presses "1" which stops his clock and starts Black's. Black then replies with a move and presses "0". Tournament rules usually require you to use the same hand to move the pieces and to press the clock.

Blitz chess

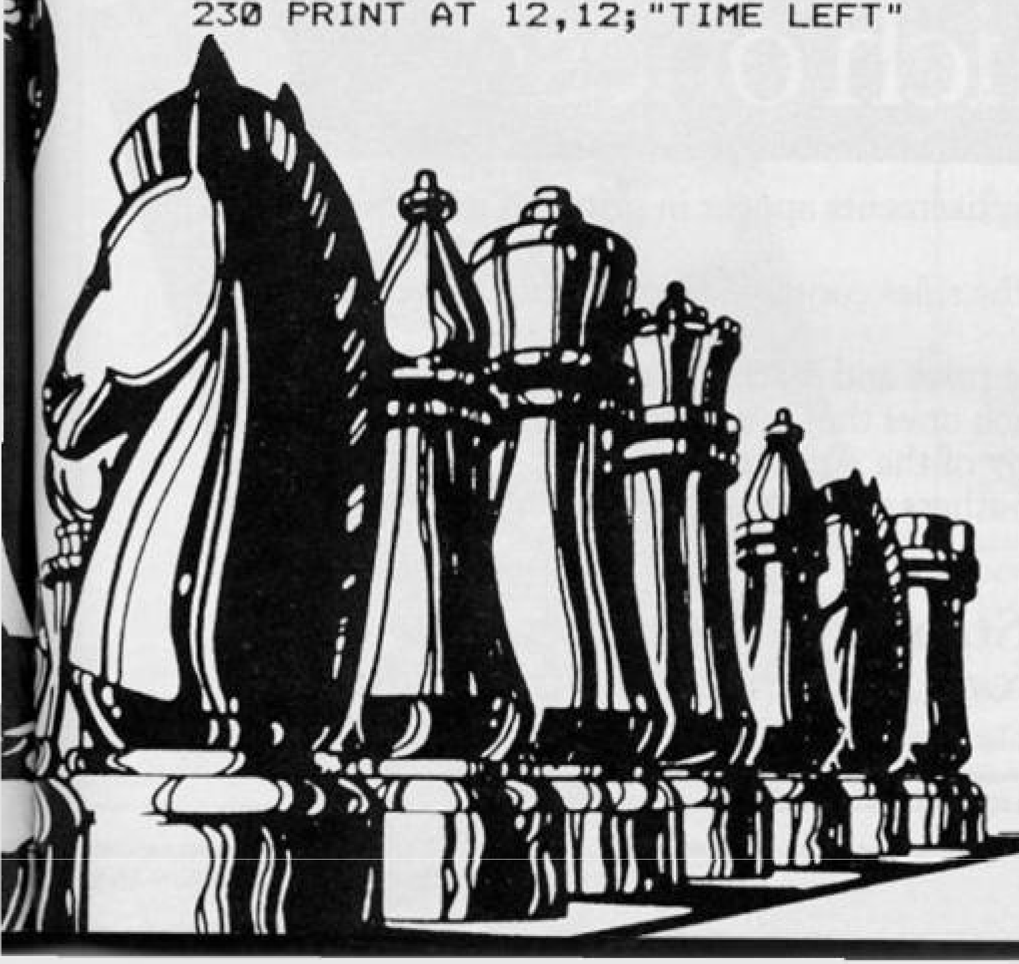
This popular form of chess, in which each player must make all his moves in, say 10 minutes can be played by entering a high move-rate ie 1000 per hour. The player who first uses up his time is the loser, unless mate has been reached first, Blitz is very good for sharpening your reactions and for improving your sight of the board.

```
   5 REM "CHESSCLOCK"
  10 CLS
  12 PRINT "ENTER MOVE RATE IN M
OVES/HOUR"
  14 INPUT X$
  16 IF X$="" THEN GOTO 14
  18 IF CODE X$<29 OR CODE X$>37
 THEN GOTO 14
  20 LET X=VAL X$
  22 PRINT X;"MOVES PER HOUR"
  30 PRINT
  32 PRINT "HOW MANY
MINUTES TO TIME CHECK?"
  34 INPUT Y$
  36 IF Y$="" THEN GOTO 34
  38 IF CODE Y$<29 OR
 CODE Y$>37 THEN GOTO 34
  40 LET Y=VAL Y$
  42 PRINT Y;" MINUTES TO
TIME CHECK"
  50 PRINT
```



```
 52 PRINT "HOW MANY MINUTES TO
BE ADDED AT EACH TIME CHECK?"
 54 INPUT Z$
 56 IF Z$="" THEN GOTO 54
 58 IF CODE Z$<29 OR CODE Z$>37
 THEN GOTO 54
 60 LET Z=VAL Z$
 62 PRINT Z;" MINUTES EXTRA AT
EACH          TIME CHECK"
 70 FOR N=1 TO 100
 72 NEXT N
 73 PRINT
 74 LET XX=INT (X*Y/60)
 75 IF X*Y/60>XX THEN LET XX=XX
+1
 76 PRINT "EACH PLAYER MUST MAK
E ";XX;" MOVES    BEFORE FIRST PE
RIOD ENDS"
 78 PRINT
 80 PRINT """S""  TO START"
 90 IF INKEY$="S" THEN GOTO 100

 92 GOTO 90
100 CLS
110 LET HW=INT (Y/60)
120 LET HB=INT (Y/60)
140 LET MW=Y-HW*60
150 LET MB=Y-HB*60
160 LET SW=0
170 LET SB=0
180 LET XW=XX
190 LET XB=XX
200 PRINT ,,,,"            ";XX;" MOVE
S IN ";Y;" MINUTES"
210 PRINT ,,,,"            WHITE
    BLACK"
220 PRINT ,,,,"            MOVES STI
LL TO MAKE"
230 PRINT AT 12,12;"TIME LEFT"
```



```
240 PRINT AT 14,5;"▛▀▀▀▀▀▜";A
T 15,5;"▌      ▐";AT 16,5;"▙▄▄
▄▄▄▟"
242 PRINT AT 14,19;"▛▀▀▀▀▀▜";
AT 15,19;"▌      ▐";AT 16,19;"▙
▄▄▄▄▄▟"
250 PRINT AT 10,8;XW;AT 10,22;X
B
252 PRINT AT 15,6;HW;":";MW;" :
";SW;AT 15,20;HB;":";MB;" :";SB
260 PRINT ,,,,,,"""1"" AFTER WH
ITES MOVE","""0"" AFTER BLACKS M
OVE","""S"" TO START"
270 IF INKEY$="S" THEN GOTO 290
272 GOTO 270
290 PRINT AT 6,7;"TO MOVE"
292 GOTO 350
300 LET XB=XB-1
305 IF XB=0 THEN GOSUB 800
310 PRINT AT 10,22;XB;" ";AT 6,
7;"TO MOVE";AT 6,21;"       "
350 PRINT AT 15,10;":";SW
355 IF SW<10 THEN PRINT AT 15,1
2;" "
360 IF SW=0 THEN GOTO 380
365 IF INKEY$="S" THEN GOSUB 90
0
370 LET SW=SW-1
371 PRINT AT 21,0;"""S"" TO STO
P                    "
372 FOR N=1 TO 9
373 IF INKEY$="1" THEN GOTO 400
376 NEXT N
378 GOTO 350
380 LET SW=59
382 IF MW=0 THEN GOTO 390
383 LET MW=MW-1
384 FOR N=1 TO 8
386 NEXT N
387 PRINT AT 15,8;MW;" "
388 GOTO 350
390 LET MW=59
392 IF HW=0 THEN GOTO 500
394 LET HW=HW-1
396 PRINT AT 15,6;HW;":"
397 PRINT AT 15,8;MW;" "
398 GOTO 350
400 LET XW=XW-1
410 PRINT AT 10,8;XW;" ";AT 6,7
;"       ";AT 6,21;"TO MOVE"
420 PRINT AT 17,7;"
              "
450 PRINT AT 15,24;":";SB
455 IF SB<10 THEN PRINT AT 15,2
6;" "
460 IF SB=0 THEN GOTO 480
465 IF INKEY$="S" THEN GOSUB 90
0
470 LET SB=SB-1
```

```
471 PRINT AT 21,0;"""S"" TO STO
P        "
472 FOR N=1 TO 6
473 IF INKEY$="0" THEN GOTO 300
474 PRINT AT 21,12
476 NEXT N
478 GOTO 450
480 LET SB=59
482 IF MB=0 THEN GOTO 490
483 LET MB=MB-1
484 FOR N=1 TO 8
486 NEXT N
487 PRINT AT 15,22;MB;" "
488 GOTO 450
490 LET MB=59
492 IF HB=0 THEN GOTO 600
494 LET HB=HB-1
496 PRINT AT 15,20;HB;":"
497 PRINT AT 15,22;MB;" "
498 GOTO 450
500 CLS
510 PRINT ,,,,,,,,"           BLACK
WON ON TIME"
520 GOTO 650
600 CLS
610 PRINT ,,,,,,,,"           WHITE
WON ON TIME"
650 PRINT ,,,,,,"""E"" TO END",
,,,"""B"" TO BEGIN ANOTHER GAME"
,,,"""R"" TO RESET CONDITIONS"
660 IF INKEY$="E" THEN GOTO 999
9
665 IF INKEY$="B" THEN GOTO 100
670 IF INKEY$="R" THEN RUN
680 GOTO 660
800 LET XB=XB+INT (X*Z/60)
810 LET XW=XW+INT (X*Z/60)
820 LET MB=MB+Z
830 LET HB=HB+INT (MB/60)
840 IF MB>=60 THEN LET MB=MB-(I
NT (MB/60))*60
845 PRINT AT 17,7;"TIME + MOVES
ADDED"
850 LET MW=MW+Z
860 LET HW=HW+INT (MW/60)
870 IF MW>=60 THEN LET MW=MW-(I
NT (MW/60))*60
875 PRINT AT 10,8;XW;" ";AT 10,
22;XB;" "
877 PRINT AT 15,6;HW;":";MW;AT
15,20;HB;":";MB
880 RETURN
900 PRINT AT 21,0;"""R"" TO RES
TART"
910 IF INKEY$="R" THEN RETURN
920 GOTO 910
```

The ZX81 soft selection

The ZX81 soft selection

ZX81 software reviewed by Nick Pearce.

Micromouse Goes De-Bugging
M C Lothlorian

Micromouse is an excellent fast moving arcade game for the ZX81. It is written entirely in machine code, and the display is in hi-resolution graphics without the need for any additional hardware.

The hi-res graphics give a screen display which is more effective than can be achieved using the graphics characters in the ZX81 ROM, and the machine code programming enables a fast responsive game. High resolution programming is becoming quite common now and can be used to particularly good effect in arcade type games, it is certainly breathing new life into many a trusty ZX81. Indeed one wonders whether the limit of the machine potential has been reached, or whether further innovative programming might lead to still better software. The ZX81 seems far from being on the shelf.

When the program is complete it will run and you progress to the next level. To make things difficult there are bugs which move around the program stealing letters, but which you can destroy using "data kill", (but Micromouse cannot carry both letters and data-kill). If more than a certain percentage of the program's letters are stolen, there is a system crash and you have to start again — this usually happened to me. It is not an easy game, but perseverence can lead to a long game and a high score.

Micromouse is a first-class high-resolution arcade game.

M C Lothlorian is at 56A Park Lane, Poynton, Stockport, Cheshire SK12 1RE



Bears In The Wood, and Climber
Unicorn Micro Systems

Back to normal ZX81 graphics in these two cassettes from **Unicorn**. Firstly, **Bears**, which resembles a maze type arcade game. Your task is to assist the last remaining grizzly to get as many of her cubs as possible, and provisions (in the form of honey), to safety in a cave. As the bear moves through the forest collecting honey the cubs occasionally become lost and have to be rescued before hunters, who roam the forest, reach them.

Instructions can be called up at the start of the game, which at first sight does seem complicated until you start to see it as a maze type game — the trees being the maze walls, hunters being ghosts, and so on. Care needs to be taken not to get the bear trapped at the edge of the forest between her cubs and the trees. Action is reasonable at the faster settings.

The game has some nice touches, the player can define his/her own movement keys, the skill level can be altered to any one of six levels, and there is a hall of fame to record the best score.

The second release from Unicorn is **Climber**. Also similar in some respects to a maze game, but perhaps better described as a computer version of snakes and ladders. "Plasma bolts" and "guardian droids" are out to get you rather than snakes, and there are power batons to collect as you climb to the top. Like Bears there is an option for control key redefinition, and speed and difficulty can be set to any of nine levels.

Both Bears and Climber are reasonably good games.

Unicorn Micro Systems is at 312 Charminster Road, Bournemouth

Extended Basic
F J G Beniest

Extended Basic is designed to overcome the limitations of Sinclair Basic on the ZX81. It enables some of the statements, found in conventional Basic but missing from the ZX81, to be used.

READ, DATA, RESTORE, DRAW, UNDRAW, IN, OUT and FILL are all provided. PRINT, CLS and SCROLL functions are included and work very much faster than their Sinclair counterparts, and utilise the whole screen too.

The interpreter is in 2.2 K of machine code situated in Basic line 0. The software is simplicity itself to use. No RAND USR statements or POKE commands. The extended Basic statements are written in REM lines and a simple GOSUB 0 causes the interpreter to execute all the new statements in the next REM line.

The REM lines can contain an unlimited number of extended Basic statements, separated by the usual ":", together with the necessary variables. Indeed, extended Basic REM lines look identical to multi-statement lines in conventional Basic.

The program is pretty well crash proof, any errors in the extended Basic statements are clearly identified by comprehensive error codes. The only problem I had was remembering to get back to single keystrokes after spelling out extended Basic statement REM lines.

The extended Basic graphic functions (DRAW, FILL, etc) are impressively demonstrated in two demo programs included in the cassette. The extended IN and OUT statements offer exciting possibilities. It should be possible to access any hardware directly and very quickly, (eg disks, sound generator, joysticks, measuring equipment, etc.).

Mr Beniest's English is far superior to my Dutch, and he must be forgiven for lapses in grammar in the instruction sheets that accompany the program. With a little thought they can be deciphered.

Extended Basic is a first class program and excellent value. It could very well become the regulation programming enhancement for the ZX81.

Extended Basic costs £4.00 from F J G Beniest, Linnaeusstraat 16, Den Haag, Holland

Racehorse Trainer and Adventure
G Barker

Racehorse Trainer is a very good simulation of a flat horse racing season on the ZX81. The object is to train your fine thoroughbred racehorses to success by entering them in races over a 20 week period, and make a profit in the process if you can. You compete with 15 other top trainers owning a total of 35 horses.

Each horse is rated on a number of factors; breeding, optimum race distance, stamina, fitness etc, and after a race these attributes are updated according to race performance. You can list a horse's rating at any time, and it is a complex matter deciding on the horse most likely to do well in each event. There are other complications too. For example, cost must be borne in mind. You start with £25,000 but weekly expenses for stabling, food, vets fees and so on, can be onorous, and you have to pay race fees, jockeys expenses etc.

The better the jockey you hire for a race the better your chances, but the more he will cost you. Horses are occasionally offered for sale by other trainers, so you need to keep a careful eye on form, and you can sell your own horses too. Disasters can sometimes occur. My horses were generally pretty ropey, didn't perform well in races even with L Piggott in the saddle, and on one occasion a horse just died between races.



The race itself is displayed, the horses in the race moving from left to right across the screen. Not a particularly inspired display, but even the chunky ZX81 graphics can get the adrenalin flowing as you shout your donkey on.

The program is menu driven and works very well. A game can last a long time, one partially completed can be saved and continued later.

Racehorse Trainer is a thoroughly enjoyable simulation and excellent value for money.

The second cassette contains three good quality adventures. **The Challenge** and **Haunted House** use both graphics and text. **In Search of the Black Staff** is a text only adventure. All three are absorbing with plenty of objects and locations in the tradition of role playing computer adventures. Again excellent value for money.

Racehorse Trainer costs £2.60 and Adventure costs £3.20 from Gavin Barker, 12 Fleming Field, Shotton Colliery, County Durham.

Readers' Reviews



Taking pen in hand, you tell us what the public really think about the goods on sale.

PYJAMARAMA by MIKRO GEN Richard Oakley

I bought this one because I liked **Automania** which was the first in the Mikro Gen's series featuring their character "Wally Week" and I thought that even if it was half as good it would be worth the money.

I was amazed to find that instead of being a quick and inferior copy of the first it was *even better*, and the graphics are fantastic!

This program is one of the wander around and jump over, dodge, jump up and collect things variety, but what makes it one of the best is the large number of detailed graphics. The rooms are not simply areas with platforms in but are fully furnished in great detail.

There is a lot of humour in the game apart from Wally's gormless expression. Try the games for example, where you have to fire knives and forks at descending chickens in space invaders style game.

The plot is quite simple. You are Wally's sleepwalking alter ego and you have to find the alarm clock and wind it up so that Wally will wake up in time for work. Various objects will try and prevent you, and all the time you are running out of "snooze energy" which is represented by a glass of milk. Collect some of the food lying around to restore this energy.

You can only carry two objects at a time and some objects need to be carrying another particular object before you can collect or use them.

The keys are responsive and easy to use and you can also define your own or use Sinclair or Kempston joystick option. The program uses O and P for left/right and M to jump. Mikro Gen's fast load system loads the program without any problem.

There is a program which I would go so far as to say is a must for any games player and is a classic of it's type. I have some of Mikro Gen's earlier programs and they were pretty ordinary — they really have improved their standards recently. I will be looking out for further releases from them.

TRANSYLVANIAN TOWER RICHARD SHEPHERD SOFTWARE Dermot Reidy

I am glad I didn't buy this game but got a present of it. It would have been a shame to waste good money on it. After having played so many good and enjoyable games on my Spectrum, this one was the worst offering I have yet seen and must be a new low for Spectrum software.

You begin the game by finding yourself in Count Kreepie's castle. You have to find your way through five mazes, kill Count Kreepie and find his treasure trove. Along the way, you have to kill man-eating, blood-sucking bats. A certain amount of bats have to be killed before you can move onto the next maze. Sounds interesting, doesn't it?

When you first load the program, there is no screen, and thus you have no nice picture to look at. There is nothing wrong with this and it does save time when you're loading your program. The instructions in the program are quite good. Then, the worst happens. You have to wait approximetly two minutes before you move into the first maze. The program is generating the maze during this time. After getting through a maze, you have to wait another two minutes before you get onto the next maze. And I thought computers were supposed to speed things up (more fool me).

The maze isn't too difficult to get through and a map can of it can be got each time you kill a bat, but the game draws the map line by line and this takes about 20 seconds. Once again, you're sitting down, twiddling your thumbs. The graphics of the maze are pretty ordinary and at times, the windows on the side wall are out of perspective, giving the program a very amateurish look. You only get one life and so if you're killed, you have to wait another two minutes before you can start again.

If you're a person with plenty of time to be looking at blank computer screens, this game is for you. Transylvanian Tower also probably suits players with a heart condition and is liable to be a good way to put the kids to sleep at night. But if you're a space invader/Krazy Kong fan, stay well away from this.

Transylvanian Tower is priced at £6.50 and comes from Richard Shepherd Software, Elm House, 23-25 Elmshott Lane, Chippenham, Slough, Berks.

INTERMEDIATE MATHS 1 ROSE SOFTWARE Jim Lock £4.95

Intermediate Maths 1 by **Rose Software** consists of six programs dealing with long multiplication, long division and fractions. The program package is attractively cased with an inlay card which gives the potential purchaser a clear description of each program's contents.

The six programs are recorded separately, three to each side of the cassette. Loading a proram also loads, line by line, a screen showing the producer's "Tudor Rose" logo in high resolution and a "welcome" message. You are then asked to type in your name and the current date before the program itself begins.

"Long multiplication" offers the choice of two levels of difficulty, multiplying either two 2-digit numbers or two 3-digit numbers. Instructions are available if required and it is not a good idea to skip these, unless you have had previous exprience with the program, as they show you how to enter your answers and, perhaps more importantly, how to "rub out" errors!

A sample run is then worked through line by line and your first question appears. You use the screen display in exactly the same way as a pencil and paper when working out the answer, entering each line of working, one number at a time, starting from the righthand side. If you make an error, pressing "R" will erase the last number printed. When each line is completed, you press "Enter" and when all your lines have been displayed, the program prompts you to add them all up and enter the final answer, again starting from the right.

The program thoughtfully includes a "Help" facility for those who are really stuck — pressing "H" enables you to see the whole sum worked out in immaculate fashion. It will also do this if you answered incorrectly.

Your score so far, of correct and incorrect answers, is shown and the option of continuing with the program is offered. Having opted, an animated "reward sequence" of a five-pin bowling alley is shown. The more correct answers you have scored, the greater the number of pins knocked down. Five pins ▶

down wins you a "strike" and a congratulatory tune.

The second program is "Long division" and they are not kidding — the worked example is 8112 34! As with the previous program, answers and working are entered exactly as you would write them. To save you losing your place in the sum, after each phase of working is completed, the program brings down the next number for you. There is no messing around with decimals — answers are given as whole numbers plus remainder.

The reward sequence for this program shows a rocket being launched at them oon, and you only need three correct answers to succeed.

The next two programs demonstrate, with well worked examples, how to use prime numers to calculate the highest common fator and lowest common multiple of a series of fractions. Both programs incorporate the useful option of entering rough working, in several stages if necessary. The usual "Help" facility is available and if used, will show you all the prime number combinations and then leave you to enter the answer.

As with the first program, five correct answers are rewarded with a complete sequence of animated graphics and sound, while less than five correct means correspondingly less of the sequence shown. With the "factor" program, you get a train crash (!), and with the "multiple" program, a caterpillar reaching some food, growing fat and turning into a butterfly.

The final two programs on the cassette are "Fractions 1", which tests your ability to add and subtract fractions, and "Fractions 2", which deals with their multiplication and division. The instructions for both programs explain the terms "numerator" and "denominator" and remind you to express your final answer in the lowest possible terms. The division sign used in the instructions looks rather odd, being made up from the Spectrum character set. In the questions themselves however, a user defined graphic is used, which looks much tidier!

The usual worked examples are provided at the start of each program, and the "enter working" and "help" facility are included. The reward sequences are, respectively, an archery target and moving arrow (which was obviously loosed by Robin Hood), and a repeat of the bowling alley seen previously.

With any of the six programs, opting not to continue at any stage prints a certificate of achievement on the screen, showing your your name, the date, program name and your percentage score. If you have been foolish enough to show your offspring or pupils how to use the printer, you run the risk of finding all your expensive rolls of paper turned into scholastic records!!

The programs in "Intermediate Maths 1" have been well designed to provide a real challenge to the user. It is a pleasure to see software which sets it's sights higher than the elementary "table-tester" and "simple sum" level. Careful attention to program detail is demonstrated by the inclusion of help and erase facilities, while the option of entering working is an original and most concept. The reward sequences all make good use of the Spectrum's sound and graphics capabilities and are great fun.

At only £4.95 for over 100K of soundly based and well produced programs, this cassette offers almost unbeatable value for money. I cannot recommend it too highly.

World Cup Football
Artic Computing
Colin Young

For those of you who cannot wait two years for the World Cup, Artic's latest release is ideal. Called **World Cup Football** it is for the 48K Spectrum and costs £6.95. Up to 9 people choose from a pool of 40 teams from England or Brazil to Tasmania and the Phillipines.

World Cup Football is the first of its kind for the Spectrum. It is unlike the strategy type football games eg. "Football Manager", it is more like an arcade game, as you control the players.

On selecting world cup mode from the initial menu, you are given a choice of the number of people taking part (1-9) and the teams each will represent. Then the first round begins. The teams run onto the pitch and take up their positions to the tune of "Match of the Day". There are only six a side which although unrealistic, is less crowded.

The game starts when player one's Centre-Forward runs up to take the centre kick and the whistle blows. You control the player, on your side, nearest the ball, who turns a different colour so he is easier to recognise. One problem is that you lose control of that player if another in your team gets closer. But, once you are used to this, the teams are easy to control using either the keyboard or a joystick (Kempston or Sinclair).

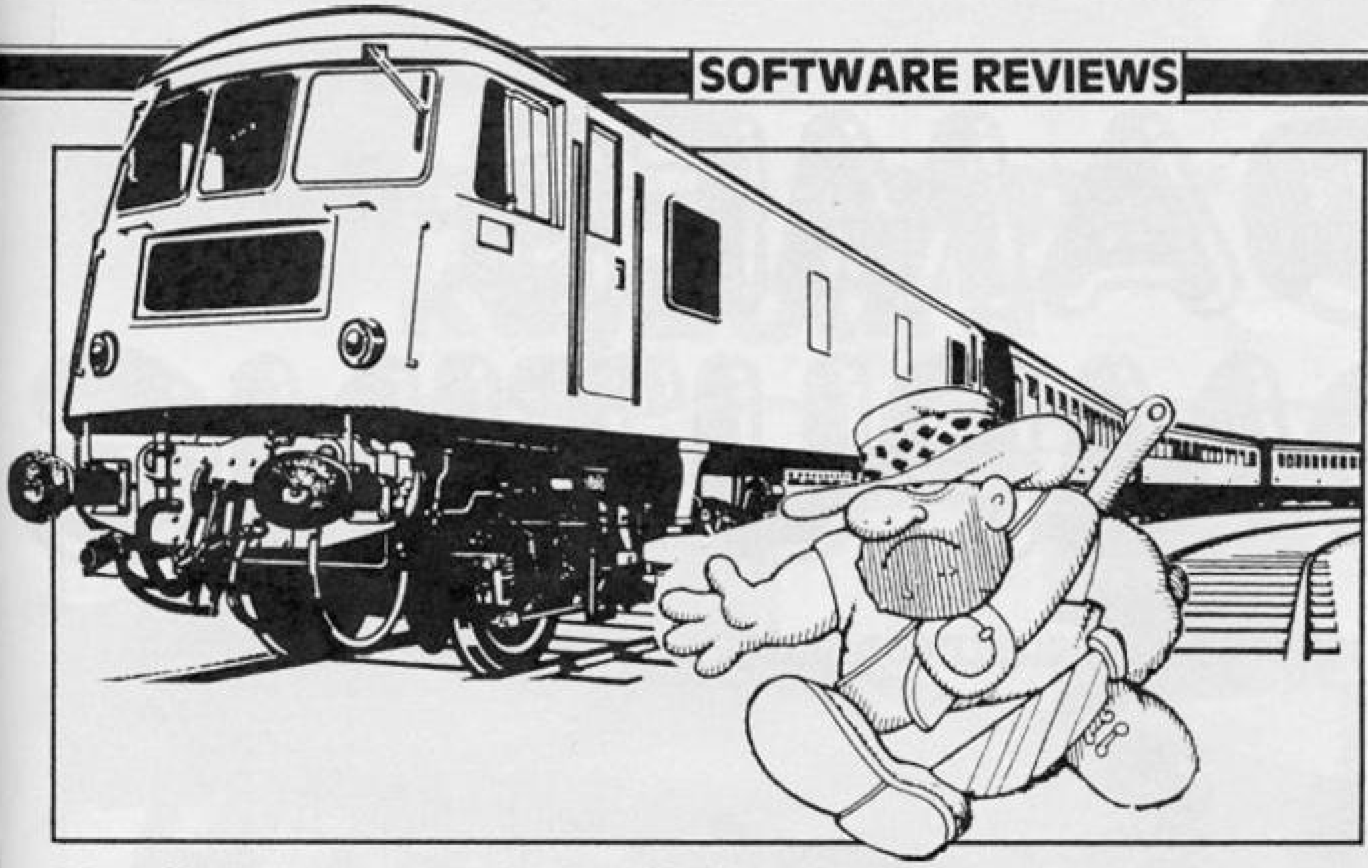
The animated running action of the players is good but unfortunately there are problems with the clashing of the colours if players overlap, leading to a confusing mess. This is not really a fault of the program, more a fault of the computer. The screen scrolls left and right with the play, the whole field being about three screens in length. A clock in the corner counts down from 90 minutes. Each minute lasts 4-5 seconds, giving a total playing time of about 7 ½ minutes.

Throughout the game the computer plays different football songs, "You'll Never Walk Alone", "We're Forever Blowing Bubbles" and, when someone scores, "Nice One Cyril". These can be turned off thankfully. In the background the crowd in the stand wave their arms and at Half-time the teams change ends while Artic adverts scroll along the top of the stand.

Throw ons and goal kicks are taken automatically and apparently aimlessly. There is also a practice mode for one or two players.

In conclusion, the game is a little easy when playing the computer and even easier when playing little sisters! However if you can find someone with some enthusiasm to play against it will be more exciting. There is more to it than just winning the cup. For example it would be easy to set up a league with a few friends using the practice mode game. If you are not keen on football I would not recommend this game but if you do enjoy football and it is raining outside, "World Cup Football" is well worth the money.





Stop the Express
Sinclair
S. Garner

This is my review of "Stop the Express"
I hope I finish it before the press.

You are an agent on the ITA train,
To get to the front is your aim.

Across the train you duck and jump,
To stop it get to the very front.

Your enemy is the evil redman,
If their daggers hit you, you are a dead man.

Only ten more carriages to go,
Watch out, help, whoa!

Over the gaps you must do jumps.
If you want to come up trumps.

Against the redman, take my word,
It helps to use the great Snakebird.

You throw this bird across the train,
To kill the reds in your domain.

With one swift throw it chops off their feet,
They fall on the track just like mincement.

Running from an evil Redman,
A passing stanchion hits my headman.

After ten carriages have rolled by,
Into the train you must comply.

No help from the snakebird is given here,
So the evil Redmen you must beware.

The time it takes a second to tick'
Is how to time your jumps and kicks.

How do you kill the evil reds,
You must kick them on their heads.

After you have stopped the "ITA" train,
Back toe beginning you go again.

Redmen thrown from the train score points,
But to do this you need double joints.

This game is rather quickative,
Thats why I think its addictive.

The graphics and colour are really the flicks
And the sound of the train goes clickety click.

This game is produced by Sinclair,
At a place far away, I know not where.

The K is O K at 48K
And it came out just after May.

The price is a mere £5.95,
And to get it all you need do is drive,

To your local computer store,
Usually on the second floor.

I'm sure they will have it in no time at all.
So it can drive you all up the wall.

Thank you for reading my silly verse,
From Garner, Stephen of course.

The 1K Games pack by ARTIC Computing £6.00
Brian Owen

This cassette consists of 11 'games', although I use the term loosely. They can be summarized by the word 'rubbish', but I will go through them one by one.

The first game, **Slot Machine,** is a crude representation of a 1p-a-go fruit machine. You pay 1p for it to pick 3 numbers and display them in the windows of the machine, which doesn't appear until you press newline to insert the first 1p. The handle stays stationary as the numbers change. You win by either getting 3 in a row, or by the numbers adding up to 7. It either announces you're bust, or, if you've got a total of 25p you will have bust the bank. **This is printed about 10 times and then the machine runs out of memory.**

Game 2 **Art**, is a familiar program where you plot a picture by moving either a PLOTting or UNPLOTting cursor around. I hadn't got far with mine, when it ran out of memory.

Game 3, **Slalom**, is where you have to move a small representation of something through a gate, represented by two o's and as many times as you can. It does known when you've crashed, however.

Catch me if you can is a game where you have to guide an 'X' onto a randomly moving 'O' that doesn't know where it is.

Space Pirate, game 5, is where you have to guide a 3-character 'ship' which descends from the top onto a randomly moving ship at the bottom, but you have to get it exactly right. You are told how much loot you took, and after 10 dockings or misses, you are told your total, and you move on to the next game.

Spacefire 1 is where you control an arrow on the left and try to shoot the randomly moving ZX81 with your laser, but you can only fire when it doesn't fire at you.

Game 7, **Spacefire 2** is exactly the same except the ZX81 moves from right to left towards you each go, and if you don't shoot each other by then, it returns to the right and tries again.

Car Crash is a substandard car driving program where you have to control an inverse V through a constant width road without hitting the edge or a V coming the other way. You can only see a fraction of the road at a time. When you crash, it multiplies the number of SCROLLS it made by 100 metres and tells you how far you drove.

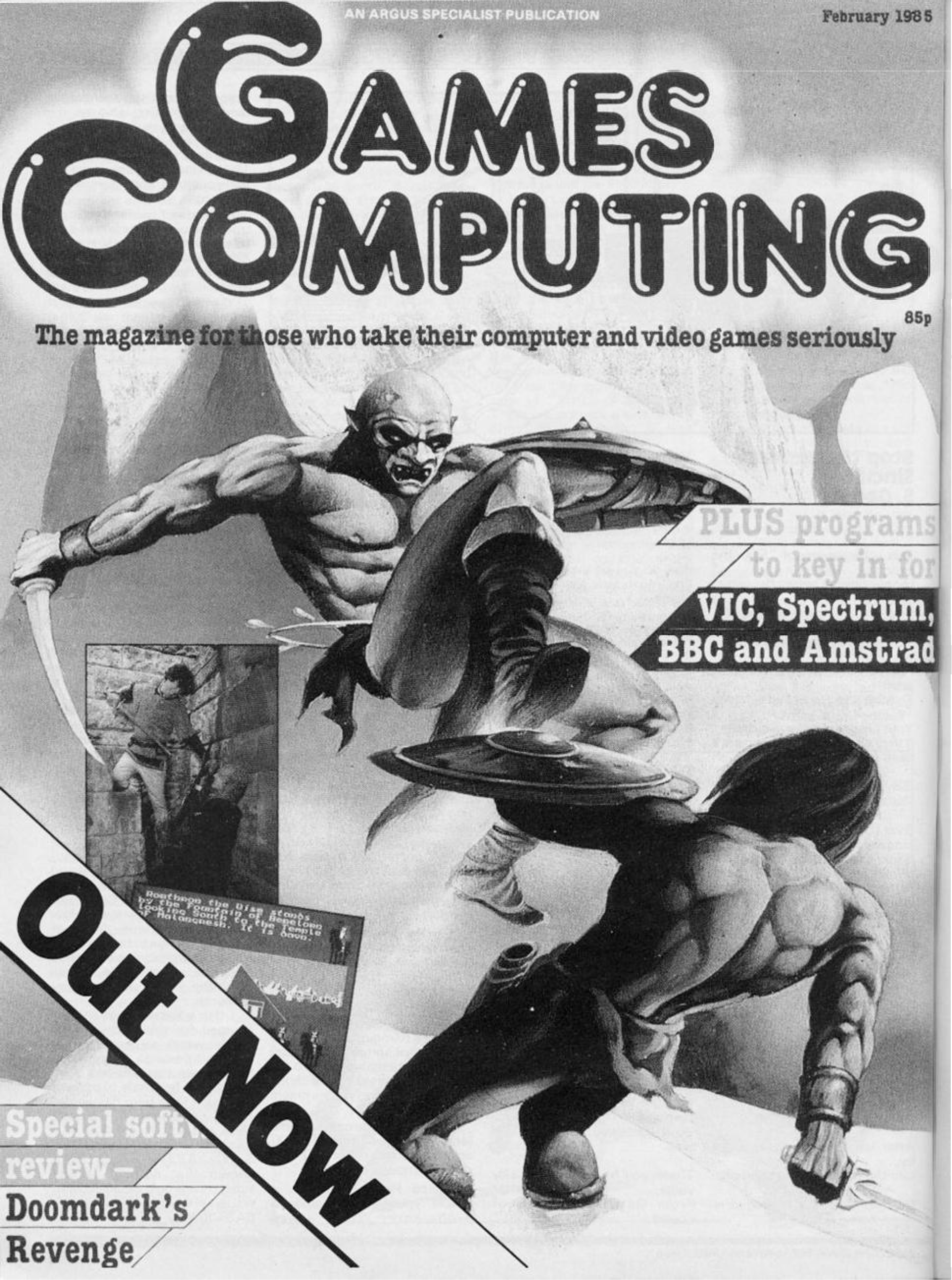
Man-eating budgies is one of the few games where the whole screen is in permanent use. You have to crash into 'O's without crashing into any of the weird 3 character shapes that are nothing like budgies.

In **Maze**, you have to move in your own time through successive lines of random blotches/spaces, until you have either hit a blob or gone through 50 blocks. You can either move horizontally, or move down another line, or fall asleep.

Program 11, **Wall**, is a primitive attempt at 'Break-out'.

Of the games, only one autoran, at least two ran out of memory, and they all had trouble listing. Games 3, 4, 5, 6 and 7 were all made worse by the fact that there was a CLS statement in the main loop. How crude!! There were however two good things to be said for all of the programs: firstly, they all loaded ok, and secondly, they all used the cursor keys — my favourite.

I don't know about the £6 charged for this package; it's hardly worth sixpence and it wouldn't come as a surprise to me to learn that this has been withdrawn from sale to prevent too much long term damage to ARTIC computing. It is a complete mystery to me how this found its way onto the shelves. Surely ARTIC would have known that nothing of any substance would be possible to write within 1K of ZX81 BASIC!!!


AN ARGUS SPECIALIST PUBLICATION
February 1985
GAMES COMPUTING
85p
The magazine for those who take their computer and video games seriously
PLUS programs to key in for
VIC, Spectrum, BBC and Amstrad
Out Now
Special soft
review –
Doomdark's Revenge

The Saga 1 Emperor Keyboard



Keyboards are now available in many shapes and sizes for the Spectrum, each having their own advantages and disadvantages, this one has an impressive range of additional keys and a very elegant style.

It is somewhat confusing at first as some keys are not where they used to be! The top row of keys are offset one position to the right, there are two keys added between the CAPS SHIFT and the Z key on the bottom row, a dot key between M and SYMBOL SHIFT, an extra ENTER where the SPACE key was and, most importantly, a full sized space bar underneath the normal keys. Dotted around at strategic points are extra Symbol Shift keys, these are to make using the additional specific function keys more convenient.

The extra keys have not just been added at random in order to make the keyboard look good, but have been carefully selected to make using the computer easier, and I must admit that after my initial trepidation at not finding keys where I expected them, and a little time using this keyboard, it really does make it easier to use!

For example, DELETE can now be also accessed by two keys next to each other at the lower left of the keyboard, instead of stretching (or using two hands) the full length of the normal board.

The keys themselves are positive and respond nicely, the legends are supplied so I presume they expect them to wear out at some time, but a quick tip here is to *CAREFULLY*- paint each one with varnish using a fine brush. This will extend their life greatly.

The keys are slightly stepped as the board is tilted forwards but seem a little cramped. The space bar is excellent and has been balanced so that catching it on the end does not cause it to tip, miss or stick as has been known to happen on some!

Installation

Fitting the keyboard to your unit may cause some slight problems as the instructions are not as idiot proof as they could be. The top of the Spectrum's case is removed and it is fitted complete with base to the bottom half of the keyboard case. It is then bolted in place, on my instructions the act of bolting was not mentioned and I was uncertain if this was correct for quite a while. Also you have to tighten the bolts just right as my over tightening resulted in the base ending up shaped like a banana! If some plastic spacing tubes to fit over the length of the bolts had been supplied then this problem would have been cheaply and effectively eliminated.

A brief comment to the effect that early models (mine is!) may require "slight modification" to the keyboard housing translates to "use a saw to cut off a couple of bits". No real problem, it took me three minutes.

The diagrams, photocopied photographs, are not good enough and need to be commercially reproduced or presented as drawings, especially as you need to use them to position the key legends. The keyboard ribbons, the one sent already had the interface attached, but the instructions tell you to attach it, are plugged in and the top fitted and screwed in place.

In use

The unit actually stands on the original Spectrum's base and is about ¾" off the ground, overlapping at the front and sides. I thought that this overhang may cause problems but in practice none were encountered, only the most heavy handed would cause it to tip.

As it uses the original base the Sinclair Interface I will fit without any problem. The microdrive lead is just long enough to allow it to sit by the side of the keyboard and it all seems to function properly. The only reservation I have is that the keyboard may then be tilted too far forward for some people.

A numeric keypad is not included on this keyboard, and if you enter a lot of data it may be worth bearing this in mind.

Summary

Although the key functions are not as well designed as on the new Spectrum+ the quality of the keys is better and, as Sinclair Research are not making the keyboard available separately, this is an attractive buy for the "normal" Spectrum owners.

Overall a beautiful and well produced piece of equipment which will enhance not only your computing but the look of the machine as well.

Stranded!

Andy Turner, recovering from his Martian Affair, now finds himself Stranded in Hull

Oh dear, I seem to have misplaced my matter transporter, how careless can you get.

Never mind, all I've got to do is look around this planet, Thear, until I find it!

Of course, nobody could even imagine the things you will encounter; hungry meteorite craters and intelligent mushrooms are two of the more credible!

To give you some sort of a chance you have three lives and ten food units to start with — you're going to need them!

Finally this is the most user UNfriendly game I've met, as Andy says; "If the Mutants don't get you then the bad jokes will!" So take your time and type this masterpiece into your machine, it should work on both ZX81 and, with only minor changes, the Spectrum.

In case anything goes wrong here is a list of the major variables to help track it down.

VARIABLES

FOOD	— food left
LIFE	— lives left
W	— water
G	— gold
D	— Mutant dissolver left
WD	— magic wand
SS	— Spam sandwiches
RH	— Red Herrings
SCORE	— score
X&Y	— position coords.
Q$,K,N,	— gen purpose variables.

```
  10 GOSUB 200
  20 LET X=5
  25 LET Y=5
  30 LET FOOD=10
  35 LET D=0
  40 LET LIFE=3
  45 LET W=0
  50 LET G=0
  60 LET SS=0
  65 LET RH=0
  67 LET WD=0
  70 LET SCORE=0
  72 PRINT "***---NEWLINE TO CON
TINUE---***"
  75 INPUT Q$
  76 CLS
  77 GOTO 900
 200 PRINT "::::::::::::::::::::::
:::::::::::::::
               ::           INSTRUCTIO
NS             ::
               ::::::::::::::::::::::
::::::::::::"
 202 PRINT "    YOU ARE A LITTLE
LOST          SPACE MAN,YOU ARE ST
RANDED ON     A DESOLATE PLANET.
                TO ESCAPE ALL YOU
 HAVE TO DO IS FIND THE MATTER T
RANSPORTER   WHICH WILL SEND YOU
BACK TO       EARTH.
                THE YEAR IS 2020
AND I SHOULDTHINK THAT IF YOU AR
E LUCKY YOU MIGHT GET HOME IN A
CENTURY OR    THREE.
                ALL THAT IS LEFT
IS FOR ME TOWISH YOU THE WORST P
OSSIBLE LUCKAND GOODBYE.
                P.S. YOU HAVE THR
EE LIVES TO BEGIN WITH.-BYE--."
 203 RETURN
 300 GOSUB 875
 301 PRINT AT 0,0;"YOU HAVE BEAT
EN THE ADVENTURE     I HATE PEOPLE
 LIKE YOU           YOU MAKE ME S
ICK.
                    BUT I BET YOU
 CANT DO IT AGAIN."
 302 LET SCORE=SCORE+10000
 303 PRINT
 304 PRINT "YOUR SCORE WAS ";SCO
RE
 305 GOTO 450
 390 PRINT "YOU HAVE EATEN SO MU
CH THAT YOU HAVE JUST BURST,HA H
A HA          SERVES YOU RIGHT FOR
 BEING SO    GREEDY."
 391 LET FOOD=10
 395 PRINT "YOU HAVE GOT FAR TOO
 MANY LIVES,THIS WILL JUST HAVE
TO STOP SO   ILL JUST KILL YOU AN
D TAKE AWAY ALL YOUR LIVES."
 396 GOTO 400
 397 PRINT "YOU HAVE JUST RUN OU
T OF FOOD    AND HAVE THEREFORE S
TARVED HA HA"
 398 GOTO 400
 400 IF FOOD<10 THEN LET FOOD=10
 402 GOSUB 875
 403 PRINT AT 0,0;"YOU ARE NOW D
EAD YOU STUPID       LITTLE SPACE
MAN.                 IT SERVES YOU
 RIGHT."
 404 IF LIFE=0 THEN PRINT "     YO
U HAVE NO LIVES LEFT HA HA     SO
 YOU WILL STAY DEAD FOR THE REST
 OF YOUR LIFE.                 YO
UR SCORE WAS ";SCORE
```





```
 405 IF LIFE=0 THEN GOTO 450
 406 PRINT "DO YOU WANT TO BE RE
INCARNATED          Y/N?"
 407 INPUT Q$
 408 IF Q$="N" THEN GOTO 475
 409 PRINT
 410 PRINT "WAIT FOR A MOMENT WH
ILE I PASS  THE HIGH VOLTAGE ELE
CTRIC       CURRENT THROUGH YOUR
 EARS."
 411 FOR N=1 TO 200
 412 NEXT N
 413 FAST
 414 FOR N=0 TO 21
 415 PRINT AT N,0;"███████████
███████████████████"
 416 NEXT N
 417 FOR N=1 TO 100
 418 FAST
 419 FOR Q=1 TO 5
 420 SLOW
 421 NEXT N
 422 CLS
 423 PRINT AT 0,0;"I BET THAT BR
OUGHT TEARS         TO YOUR EYES,
                    YOU MAY NOW C
ARRY ON YOUR SEARCHWHERE YOU LEF
T OFF."
 424 LET LIFE=LIFE-1
 425 GOTO 870
 450 PRINT
 452 PRINT
 453 PRINT "DO YOU WANT TO START
 ANOTHER    ADVENTURE Y/N?"
 454 INPUT Q$
 456 PRINT "WAIT A MO."
 457 FOR N=1 TO 100
 458 NEXT N
 459 CLS
 460 CLEAR
 461 RUN
 475 PRINT "BYE BYE."
 476 FOR N=1 TO 100
 477 NEXT N
 478 NEW
 870 GOSUB 875
 872 GOTO 900
 875 PRINT AT 21,0;"***---NEWLIN
E TO CONTINUE---***"
 876 INPUT Q$
 880 FOR N=1 TO 22
 890 SCROLL
 895 NEXT N
 897 PRINT AT 0,0;
 899 RETURN
 902 PRINT "WHICH DIRECTION?
            N,S,E OR W."
 905 INPUT D$
 910 IF D$="N" THEN LET X=X-1
```

```
 915 IF D$="S" THEN LET X=X+1
 920 IF D$="E" THEN LET Y=Y+1
 925 IF D$="W" THEN LET Y=Y-1
 930 IF D$<>"N" AND D$<>"S" AND
D$<>"E" AND D$<>"W" THEN GOTO 90
5
 935 LET SCORE=SCORE+100
 936 IF FOOD>=30 THEN GOTO 390
 937 IF FOOD=0 THEN GOTO 397
 938 IF LIFE>=6 THEN GOTO 395
 939 LET FOOD=FOOD-1
 940 IF X=0 THEN LET X=9
 945 IF Y=0 THEN LET Y=9
 950 IF X=10 THEN LET X=1
 955 IF Y=10 THEN LET Y=1
 960 GOSUB ((X*1000)+(Y*100))
 970 GOTO 800
1100 PRINT "TREAD CAREFULLY."
1110 RETURN
1200 GOTO 2100
1300 PRINT "ITS GETTING VERY HOT
 AROUND HERE"
1310 RETURN
1400 GOTO 1500
1500 PRINT "YOUVE ENTERED A SCOR
CHING HOT   RADIOACTIVE DESERT
            DO YOU:-1)TURN BACK?
                    2)TRY TO CRO
SS IT?"
1501 INPUT Q$
1502 IF Q$="2" THEN GOTO 1510
1503 PRINT "GOOD CHOICE."
1504 RETURN
1510 PRINT "IDIOT YOU HAVE JUST
BEEN FRIED  HA HA -SIZZLE-"
1511 GOTO 400
1600 GOTO 1500
1700 GOTO 1300
1800 PRINT "I DARE YOU TO GO E."

1810 RETURN
1900 GOTO 7800
2100 PRINT "YOU HAVE JUST ENTERE
D AN AREA   WHERE THE EARTHS CRU
ST IS VERY  THIN-TREAD CAREFULLY
."
2101 IF G=0 THEN RETURN
2102 PRINT "OH DEAR THAT GOLD OF
 YOURS IS SOHEAVY THAT IT HAS CA
USED YOU TO SINK THROUGH THE CRU
ST AND FALL INTO OBLIVION -BYEEE
EEEEEEEEEEE"
2103 GOTO 400
2200 GOTO 3100
2300 PRINT "RICHES ARE AT HAND."

2310 RETURN
2400 GOTO 1300
2500 PRINT "YOU HAVE FOUND SOME
FOOD        DO YOU :-1)EAT IT?
                    2)DIG A HOL
E AND                 BURY IT?
                    3)IGNORE IT
?"
2501 INPUT Q$
2502 IF Q$="1" THEN PRINT "YUM Y
UM."
2503 IF Q$<>"2" THEN RETURN
2504 PRINT "WHILE DIGGING THE HO
LE YOU FALL IN AND DIE OF STARVA
TION HA HA  YOU WONT TRY TO BE F
UNNY ON THISPROGRAM AGAIN WILL Y
OU."
2505 GOTO 400
2600 GOTO 1300
2700 PRINT "THERES A DESERT NEAR
 AND A       FOREST FAR FAR TO S.
W."
2710 RETURN
2800 PRINT "YOU FIND A NOTE SAYI
NG          8 W. 5 S."
2810 RETURN
2900 PRINT "I DARE YOU TO GO N."

2910 RETURN
3100 PRINT "YOU HAVE FOUND A BAR
 OF GOLD    DO YOU :-1)EAT IT?
                    2)TAKE IT W
ITH YOU?            3)IGNORE IT
?"
3101 INPUT Q$
3102 GOTO ((VAL Q$*10)+3100)
3110 PRINT "BURP,THAT DIDNT DO M
UCH FOR YOURINDIGESTION."
3111 LET FOOD=FOOD+1
3112 RETURN
3120 PRINT "WISE."
3121 LET G=1
3122 RETURN
3130 PRINT "NOW WAS THAT WISE???
"
3131 RETURN
3200 PRINT "THERES A DESERT TO T
HE N.E."
3210 RETURN
3300 GOTO 7800
3400 PRINT "YOU HAVE COME ACROSS
 SOME RATHERUNFRIENDLY RADIOACTI
VE ROCKS     DO YOU :-1)PANIC?
                    2)RUN LIKE
HELL?"
3401 IF W=1 THEN PRINT TAB 9;"3)
POUR WATER ON THEM?"
3402 INPUT Q$
3403 GOTO ((VAL Q$*10)+3400)
3410 PRINT "OH DEAR THE ROCKS GO
T YOU,YOU   HAVE BEEN FRIED HA H
A SIZZLE."
```

```
3411 GOTO 400
3420 IF RND<.5 THEN GOTO 3410
3421 PRINT "YOU ARE LUCKY,YOUVE
ESCAPED."
3422 RETURN
3430 PRINT "ARENT YOU CLEVER THE
 ROCKS HAVE DISSOLVED-YOU MAKE M
E SICK."
3431 LET W=0
3432 RETURN
3500 PRINT "THERES A HINT OF MAG
IC IN THE   AIR."
3510 RETURN
3600 GOTO 6200
3700 PRINT "YOU HAVE ON YOUR SEA
RCH COME       ACROSS A VERY INTELL
IGENT          MUSHROOM...DO YOU :-
               1)PLAY A GAME OF CHE
SS WITH IT? 2)ASK IT THE WAY?
               3)EAT IT?
               4)PICK IT?"
3705 INPUT Q$
3710 GOTO ((VAL Q$*10)+3710)
3720 PRINT "THE MUSHROOM BEATS Y
OU AND TURNSYOU TO STONE FOR BEI
NG SUCH A   BAD PLAYER-SERVES YO
U RIGHT FOR BEING A CLEVER DICK.
"
3725 GOTO 400
3730 LET K=RND
3731 IF K>=.5 THEN PRINT "MUSHRO
OMS ANSWER :-                GO 4 T
IMES SOUTH                   AND TW
O TIMES EAST                 YOU MA
Y BE LUCKY                   OR A M
UTANTS FEAST."
3732 IF K<.5 THEN PRINT "MUSHROO
MS ANSWER :-                GO 2 TI
MES EAST                    AND 5 T
IMES NORTH                  AND YOU
LL FIND THE TRANSPORTER  AT LEAS
T I THINK YOU OUGHTA."
3733 RETURN
3740 LET K=RND
3741 IF K>=.75 THEN PRINT "HARD
LUCK THE MUSHROOM IS          POISO
NOUS AND YOUR DEAD."
3742 IF K>=.75 THEN GOTO 400
3744 PRINT "TASTY WAS IT?"
3745 LET FOOD=FOOD+2
3746 RETURN
3750 PRINT "YOUVE KILLED IT MURD
ERER."
3751 RETURN
3800 GOTO 7800
3900 PRINT "YOU FIND A NOTE SAYI
NG             EAST? OR WEST?
               NO SOUTH IS BEST."
3910 RETURN
4100 PRINT "THERES A FOREST TO T
HE SOUTH."
4110 RETURN
4200 PRINT "FOOD IS CLOSE."
4210 RETURN
4300 GOTO 2500
4400 GOTO 3500
4500 PRINT "YOU HAVE FOUND A MAG
IC WAND        DO YOU:-1)PICK IT UP
?                        2)THROW IT A
WAY?                     3)USE IT AS
A TOOTHPICK?             4)IGNORE IT?
"
4501 INPUT Q$
4502 GOTO ((VAL Q$*10)+4500)
4510 PRINT "VERY WISE."
4511 LET WD=1
4512 RETURN
4520 PRINT "IT DIDNT LIKE BEING
THROWN AWAY SO IT HAS THROWN YOU
 AWAY IN    SOME OBSCURE DIRECTI
ON HA HA."
4521 LET X=INT (RND*9)+1
4522 LET Y=INT (RND*9)+1
4523 RETURN
4530 PRINT "WHILE USING IT AS A
TOOTHPICK    YOU SWALLOWED IT AND
 DIED HA HA STOP TRYING TO BE A
CLEVER DICK."
4531 GOTO 400
4540 PRINT "WISE."
4541 RETURN
4600 GOTO 5300
4700 PRINT "SOMETHING INTELLIGEN
T LIES         TO THE N."
4710 RETURN
4800 GOTO 6200
4900 GOTO 7800
5100 PRINT "YOU ARE GETTING CLOS
ER TO A        FOREST."
5110 RETURN
5200 PRINT "WATERS NEAR."
5210 RETURN
5300 PRINT "YOUVE FOUND A BOTTLE
 OF WATER    DO YOU :-1)DRINK IT?
                          2)CARRY IT
WITH YOU?                 3)IGNORE IT
?"
5301 INPUT Q$
5302 IF Q$="3" THEN RETURN
5303 IF Q$="2" THEN GOTO 5310
5304 PRINT "SLURP GUZZLE"
5305 LET FOOD=FOOD+2
5306 RETURN
5310 LET W=1
5311 RETURN
5400 PRINT "YOU SEE A CRASHED SP
ACE SHIP ANDFOOTPRINTS LEADING S
."
```

```
5410 RETURN
5500 PRINT "YOU HAVE RETURNED TO
 THE START  YOU IDIOT,I THOUGHT
I TOLD YOU  TO FIND THE MATTER T
RANSPORTER."

5510 RETURN
5600 GOTO 7800
5700 GOTO 2500
5800 PRINT "YOU HAVE FOUND A FIS
H,         A SMELLY RED HERRING
 TO BE EXACTDO YOU :-1)EAT IT?
                     2)TAKE IT W
ITH YOU?             3)PUT YOUR
OXYGEN MASK           ON AND RU
N?"
5801 INPUT Q$
5802 GOTO ((VAL Q$*10)+5800)
5810 PRINT "RATHER YOU THAN ME."

5811 LET FOOD=FOOD+1
5812 RETURN
5820 PRINT "PHEW WHAT A STINK."
5821 LET RH=1
5831 RETURN
5900 GOTO 6800
6100 PRINT "THERE IS A FOREST JU
ST SOUTH."
6200 PRINT "YOU HAVE JUST FOUND
A BOTTLE OF REINCARNATING OIL,SO
 YOUVE JUST GAINED ANOTHER LIFE.
 DAMN."
6201 LET LIFE=LIFE+1
6202 RETURN
6300 PRINT "YOU ARE VERY CLOSE T
O A FOREST."
6310 RETURN
6400 PRINT "YOU SEE A LINE OF FO
OTPRINTS     COMING FROM N. AND T
URNING TO GOO."
6410 RETURN
6500 PRINT "YOU SEE A SKELETON I
N A SPACE    SUIT."
6510 RETURN
6600 GOTO 4800
6700 GOTO 3400
6800 PRINT "YOU HAVE FOUND A SPA
M SANDWICH  DO YOU :-1)EAT IT?
                     2)TAKE IT W
ITH YOU?             3)RUN AWAY
IN CASE IT            EATS YOU?
                     4)IGNORE IT
?"
6801 INPUT Q$
6802 GOTO ((VAL Q$*10)+6800)
6810 PRINT "YUK."
6811 LET FOOD =FOOD+2
6812 RETURN
6820 LET SS=SS+1
6821 PRINT "I HOPE YOURE VERY HA
PPY TOGETHER"
6822 RETURN
6830 PRINT "WELL DONE THEYRE VIC
IOUS THESE  SPAM SANDWICHES."
6831 RETURN
6840 PRINT "YOU SHOULD HAVE RAN
AWAY COS ITSJUST TURNED NASTY AN
D HAS EATEN YOU -GULP."
6841 GOTO 400
6900 PRINT "YOU HAVE FOUND THE E
VIL VIDEO    GAMES WIZARD WHO IS
ABOUT TO     TURN YOU INTO A SPAM
 SANDWICH    COS HES HUNGRY,
             DO YOU :-1)CRY?
                      2)PRAY?
                      3)OFFER HIM
 A SPAM                SANDWICH?
"
6901 IF WD=1 THEN PRINT TAB 9;"4
)USE YOUR MAGIC WAND
 ON HIM?"
6902 INPUT Q$
6903 GOTO ((VAL Q$*10)+6900)
6920 PRINT "THERE IS A LITTLE PU
FF OF GREEN SMOKE AND YOUR A SPA
M SANDWICH   MUNCH MUNCH GULP,WER
E."
6921 GOTO 400
6930 IF SS=0 THEN GOTO 6935
6931 LET SS=0
6932 PRINT "HES TAKEN YOUR WHOLE
 SUPPLY OF   SPAM SANDWICHES,BUT
HAS LET YOU GO."
6933 RETURN
6935 PRINT "HE WAS GOING TO TAKE
 YOUR        SANDWICHES,BUT YOU H
AVENT GOT    ANY AW WHAT A SHAME.
"
6936 GOTO 6920
6940 PRINT "THE WIZARD DISAPPEAR
S IN A PUFF OF GREEN SMOKE LEAVI
NG BEHIND A SPAM SANDWICH."
6941 GOSUB 875
6943 GOTO 6800
7100 PRINT "YOU ARE IN A DARK DA
MP ALIEN     FOREST-DO YOU EAT TH
E FRUIT OF   THE TREES Y/N?"
7101 INPUT Q$
7102 IF Q$="N" THEN GOTO 7120
7103 IF RND<.45 THEN GOTO 7115
7104 PRINT "YUM YUM"
7105 LET FOOD=FOOD+2
7106 GOTO 7120
7115 PRINT "IDIOT THIS FRUIT IS
POISONOUS."
7116 GOTO 400
7120 PRINT "ITS UP TO YOU."
7121 IF RND>.2 THEN RETURN
```

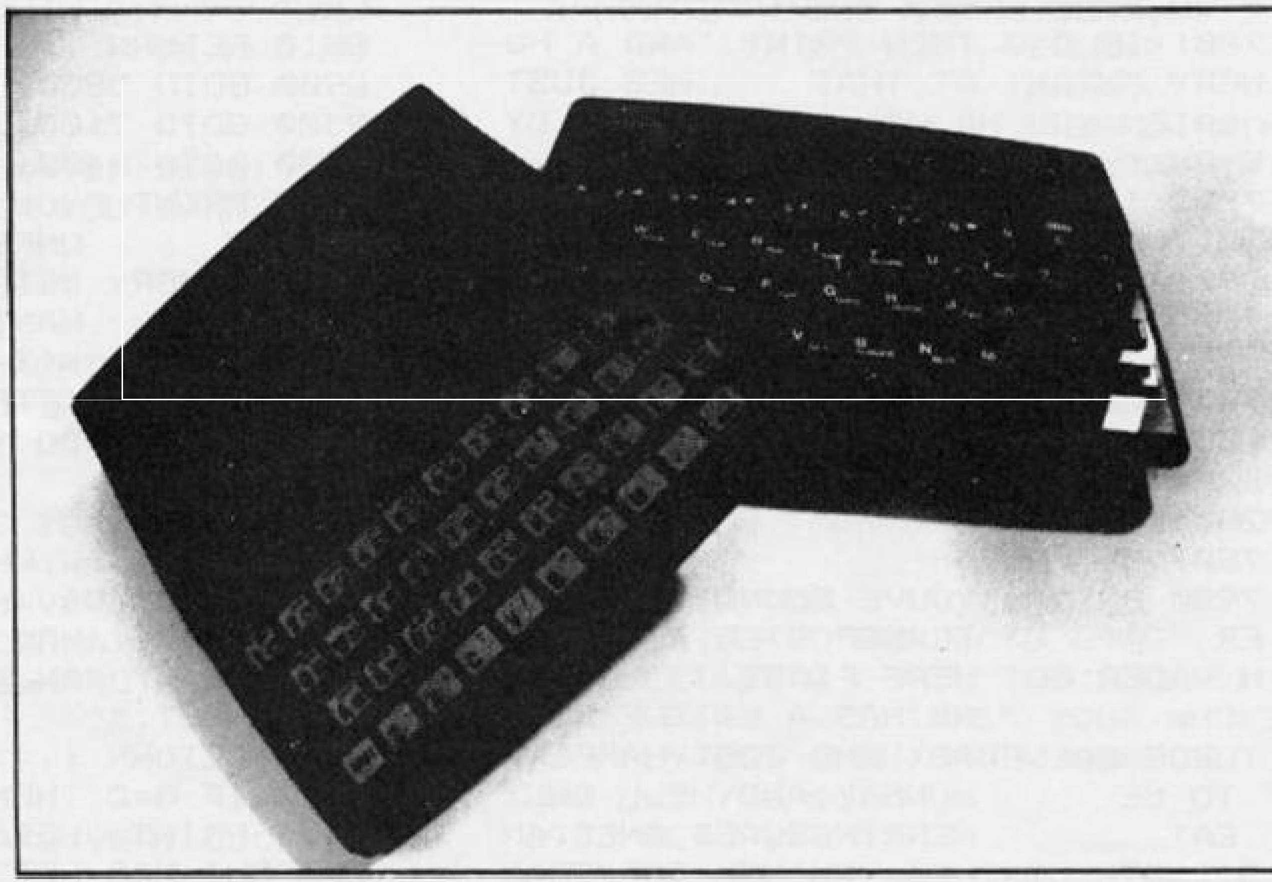
```
7122 PRINT " YOU CLUMSY FOOL YOU
 HAVE JUST  STEPPED ON THE ROOT
OF A TREE   MAKING IT VERY ANGRY
,SO IT HAS  EATEN YOU MUNCH MUNC
H GULP HA HA"
7123 GOTO 400
7300 GOTO 7100
7400 GOTO 7800
7500 GOTO 6300
7600 GOTO 3400
7700 GOTO 5300
7800 PRINT "YOUVE STUMBLED ACROS
S A RATHER  LARGE HUMANOID-WHOOP
S OH DEAR   HES A MUTANT"
7801 IF D=0 THEN PRINT "AND A HU
NGRY MUTANT AT THAT      HES JUST
 EATEN YOU HA HA          --BYE BY
E-GULP SLURP."
7802 IF D=1 THEN PRINT "YOU RAN
OUT OF SPRAY."
7803 IF D=1 THEN LET D=0
7804 IF D=0 OR D=1 THEN GOTO 400

7805 PRINT "BUT LUCKILY YOU HAVE
 SOME MUTANT DISSOLVING SPRAY AN
D HAVE USED IT."
7806 LET D=D-1
7807 RETURN
7900 PRINT "YOUVE FOUND THE MATT
ER          TRANSPORTER,BUT DART
H VADER GOT HERE FIRST.
            HE HAS A LITTLE ROBO
T DOG CALLEDREX WHO JUST HAPPENS
 TO BE      HUNGRY,AND HELL ONLY
 EAT        HERRINGS,RED ONES.AN
D DARTH WONTLET YOU USE THE TRAN
NY WHILE REXIS HUNGRY."
7901 IF RH=0 THEN GOTO 7920
7902 PRINT "DO YOU GIVE REX A RE
D HERRING?        Y/N?"
7903 INPUT Q$
7905 IF Q$="N" THEN GOTO 7910
7906 PRINT "THATS A GOOD BOY,DAR
TH HAS LET  YOU USE THE TRANSPOR
TER,SO YOUVEESCAPED,-YOU MAKE ME
 SICK"
7907 GOTO 300
7911 PRINT "DARTH HATES GREEDY P
EOPLE AND SOHE HAS KILLED YOU WI
TH HIS LIGHTSABER HA HA."
7912 GOTO 400
7920 PRINT "AW WHAT A SHAME YOU
HAVENT GOT  ANY HERRINGS,SO YOUL
L JUST HAVE TO GO AND GET ONE HA
 HA."
7921 RETURN
8100 GOTO 7100
8200 PRINT "YOU HAVE FOUND A CAN
 OF MUTANT  DISSOLVING SPRAY,YOU
 MAY USE IT TO PROTECT YOURSELF
WHEN YOU SEEA MUTANT."
8201 LET D=D+3
8202 RETURN
8300 GOTO 7100
8400 GOTO 8500
8500 PRINT "YOU CAN SEE A FOREST
            TO THE WEST."
8510 RETURN
8600 PRINT "THERES FOOD NEAR."
8601 RETURN
8700 GOTO 2500
8800 PRINT "A TERRIBLE FISHY SME
LL          FILLS THE AIR."
8810 RETURN
8900 GOTO 5800
9300 GOTO 7100
9400 GOTO 8400
9500 PRINT "YOU HAVE JUST MET A
RATHER      UNFRIENDLY AND ALSO
VERY HUNGRY METEORITE CRATER WHO
 EITHER     WANTS FOOD OR GOLD T
O BUY SOME  WITH,FROM THE VIDEO
GAMES WIZARDAFTER THIS ADVENTURE
            DO YOU :-1)RUN?
                     2)OFFER HIM

 GOLD?               3)OFFER HIM
 A SPAM SANDWICH?"
9501 INPUT Q$
9502 GOTO ((VAL Q$*10)+9500)
9510 PRINT "AW HOW UNFRIENDLY CA
R YOU GET."
9511 RETURN
9520 IF G=0 THEN GOTO 9525
9521 PRINT "HES TAKEN ALL YOUR G
OLD BUT HAS LET YOU GO."
9522 LET G=0
9523 RETURN
9525 PRINT "AW YOU HAVENT GOT AN
Y GOLD      SO HES EATEN YOU BYE
 BYE."
9526 GOTO 400
9530 IF SS=0 THEN GOTO 9535
9531 PRINT "YOUR LUCKY HE LIKES
SPAM...     HES LET YOU GO."
9532 LET SS=SS-1
9533 RETURN
9535 PRINT "WHAT A SHAME HE PREF
ERS TASTY   LITTLE SPACE MEN LIK
E YOU       SO HES EATEN YOU-GUL
P."
9536 GOTO 400
9600 GOTO 8800
9700 GOTO 5800
9800 PRINT "A FRIENDLY TIME TRAV
ELLER       SUGGESTS YOU GO N.E.
"
9810 RETURN
9900 GOTO 5100
```

De-bugger

Getting a program typed in is often only the start of your problems. Ed to the rescue.



Typing in a program is a useful exercise. Apart from the patience required, techniques learned and the end program to be used, probably the most educational part of it is tracking down the bugs introduced by yourself or occasionally by our publication system.

In debugging you gain a much deeper insight and understanding on how the program actually works than by merely typing it in, but tracking down these errors is an art in itself and needs some skill. So here are some tips to help you in your efforts when faced with that cryptic error report!

1 NEXT without FOR
Look back through the program, either the loop has not been set up — no related FOR 'letter' = No1 TO No2 line, or the letter has been re-used as an ordinary variable within the loop with a LET 'letter' = No.

2 Variable not found
This is one of the most common errors. Again, the problem may not lie in the line where the error was detected and reported. If there is only one variable, which may be one or more letters or a string ($) variable, then that is the problem. There may be more than one variable in the line section reported and you will have to identify the offending one. In a line PRINT AT Y,X;A$ the culprit could be Y or X or A$. To find out which of them is causing the problem (it may be more than one) type in turn as a direct command:

PRINT Y Enter/Newline
PRINT X Enter/Newline
PRINT A$ Enter/Newline

Note which produces the error report. Now look back through the program printout for the line which sets it up — usually a LET or FOR command. Did you leave it out? Does the program get there or has a GOTO/GOSUB been wrongly addressed?

3 Subscript wrong
Connected with DIM A(No) or DIM A$(No). If the number in the brackets on the line where the error is reported is greater than the one in the original DIM statement, is not an integer or is less than 1, then this report is generated. If the subscript — number in brackets — is a number then check and change, however, if it is a variable then follow the procedure for tracing variables. It has probably exceeded the limits, look for lines with the variable being altered with + − ✱ / : if necessary add limiting code. For example:

IF X >10 THEN LET X = 10

4 Out of memory
As well as for programs which are too big, it may happen if the previous program set RAMtop. Before despairing, enter CLEAR USR "a"-1 on the Spectrum: on the ZX81 SAVE the program, turn the machine off and on, then reload the program.

7 RETURN without GOSUB
Somehow the computer has reached a RETURN command other than via a GOSUB instruction. Check a GOTO hasn't been entered in place of a GOSUB. Check for a missing GOSUB.

B Integer out of range
An integer (whole number) either as a number of variable is too big or small and you are attempting to do something like PRINT AT 0,33 — not allowed! Check any variables involved as per report 2 and trace it back through the program looking for adjustments to it by + − ✱ / : Add limiting code if needed — see report 3

E Out of DATA
A Spectrum problem. Check the number of DATA items match the number of READs; usually one (or more) has been missed out. Attempting to reread a DATA list without first using a RESTORE command will cause this and it can happen on an auto start program (saved with a LINE number). Good programming usually RESTOREs to the correct line number before using READ.

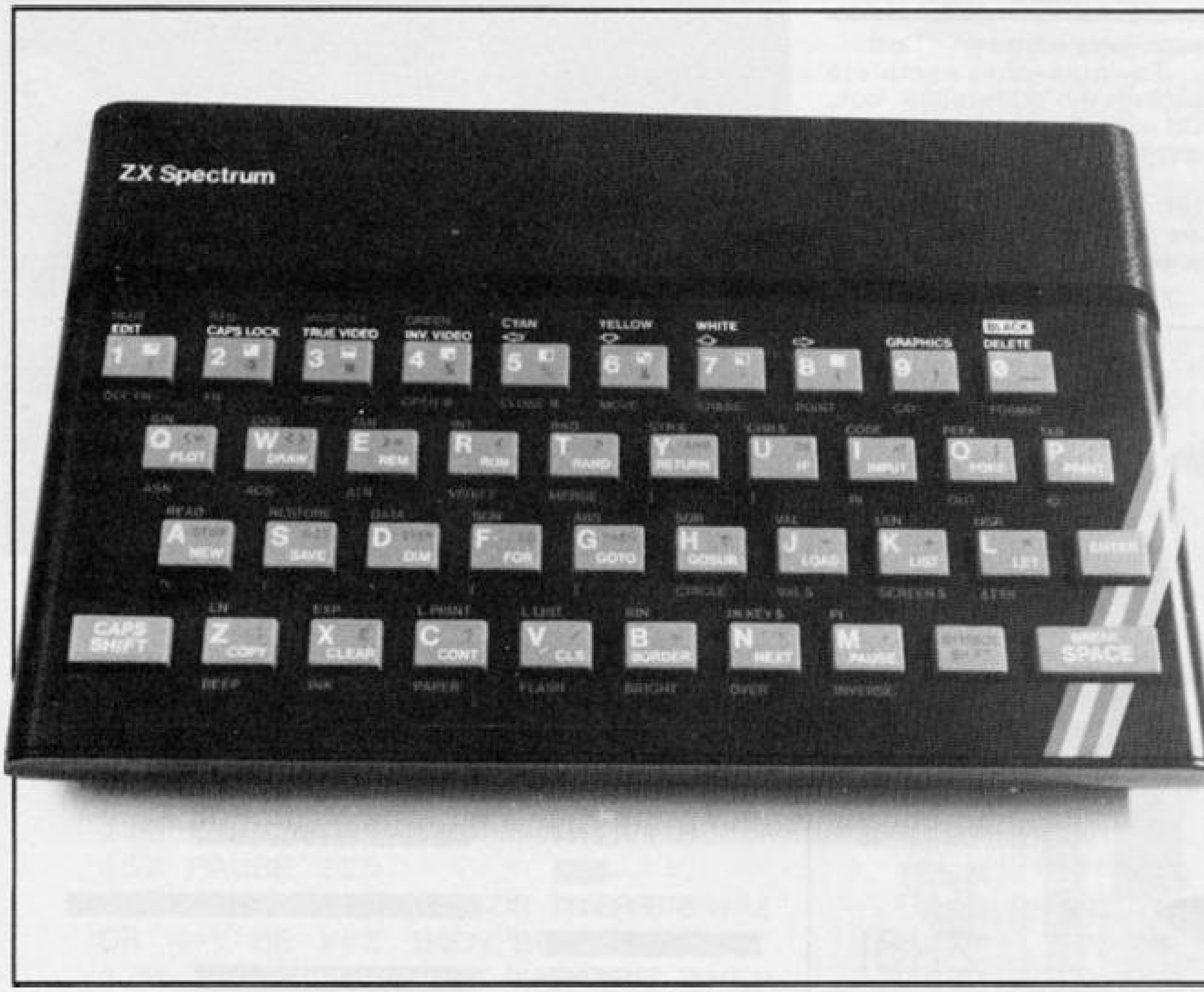
I FOR without NEXT
See report 1 but this time the NEXT is missing!

Note that the letters I have used for examples could by ANY letters not just A$,X,Y etc and depend on the particular choice of the programmer.

This is by no means a comprehensive list but I have tried to cover many of the most common error reports. Personally, I get almost as much satisfaction from debugging as I do from programming I do assure you, however, that there is absolutely no truth in the rumour that we deliberately inject bugs into our listings in order to introduce you to the dubious delights of debugging!

The Rotronics Wafadrive

Colin Christmas gets in the fast lane



HARDWARE

When fast access storage systems for the Spectrum were first introduced to us out here in User Country, everybody who could afford it went for a Drive, Disc or Micro — the members of the new elite would ask one another. What's it like? asked the rest of us taking a knife to the piggy bank.

Once again it seemed that the Spectrum was being raised to new heights. Those cassette recorders suddenly seemed to look quaint and out dated. Serious micro users whether programmers or like myself just users and consumers of software and peripherals that look as though they mean business, recognised the potential of the new hardware.

One of the newest is the ROTRONICS WAFADRIVE. It is an impressive, neat and businesslike unit weighing some 900g and occupying a pride of place 230mm × 110mm × 80mm behind the Spectrum. It is connected to the expansion port by a sturdy ribbon cable. It uses two magnetic tape drives incorporated in the unit and small compact plastic cartridges or wafers, 67 ×45 ×6mm in size. These are available with three nominal capacities of 16K, 64K and 128K.

In practical terms, the 16K wafer is most useful when speedy access to limited amounts of information, small programs, is required. Rather like a pad or notebook, a 'try out area' before the finished data or program needs to be safely filed away and stored for future reference. Here, the other wafers fulfill their most useful function. Location of files on the 64K and 128K wafers or cartridges takes longer of course. Nevertheless with a search speed of 15 inches per second, both drives in the unit offer a maximum or 'worst' access time of 45 seconds using the 128K wafer.

The infinite loop tape (one sixteenth of an inch wide) inside each wafer is completed by a conductive splice which can be 'read' by the Wafadrive System when the drives are operating. LEDs are used to indicate the status of each drive and also when the power is on in the unit. The wafers are not easily damaged, handle well and can have the information stored on them protected in the same way as normal cassettes. The novelty of rapid access, storing and retrieving programs and data, takes a long time to wear off.

The Wafadrive operating system reserves about 2K of the Spectrum's memory for the two drive directories, which hold all the information about the wafers in the drives and the files stored on them. Also for systems variables, some of which can be usefully PEEKed and POKEd and also for the read/write buffers used by the system to hold sections of programs as they are read from or written to the wafers.

EXTENDED BASIC

The operating system also provides the Extended BASIC commands which give access to all the extra facilities of the Wafadrive. Separate interfaces are not necessary with this unit. Rotronics have incorporated both the Centronics and RS232 Interfaces and along with the familiar expansion bus, ports for these are located at the rear. This means of course that the Spectrum can then drive any compatible printer. But via the RS232 Interface though in many ways unsatisfactorily slow, information can be received say from modems or even other computers.

Finally, a User Manual, a blank wafer and a word processor wafer are supplied with the unit. For those of us who choose to spend a lot of time over a typewriter keyboard producing vast amounts of material which then has to be documented and then filed, this latter is a useful facility which can be used as soon as the Wafadrive is connected up after unpacking. There again, word processor facilities have a special appeal if not function for me in any case. However, after using TASWORD for a long time, it is my opinion that it is hard to beat.

The manual is easy to use for both beginner and expert, young and old alike. I liked its' format and size, something like an office file or manuscript itself. More of a Manual than an instruction book. It contains masses of information relating to the straighforward use of the unit but at the same time offers to stretch your knowledge and experience of both the Spectrum and BASIC. Not to mention a lot of useful and well expressed information about how the unit works.

Anyone going for a Drive needs to take this newcomer into account when the day comes.

Bandit

A grasphically excellent fruit machine simulation from Ian Driver who gambles in Lancashire.

This is a program which uses several subroutines to amuse the player, the graphics have been used effectively to produce full and varied screen displays.

Prompts are given at each stage and playing is straightforward. Good luck, I found it to be reasonably accurate — I lost.

To give the complete package the 'professional' look, cut out and use the cassette insert provided.

NB. All words typed in bold ink are in inverse video, only enter space where space is written.



Bandit cassette-insert




```
1 CLS
2 PRINT "████████████████████████████████"
3 PRINT "  BET                 ██BANDIT ██"
4 PRINT "████████████████████████████████"
5 PRINT " DECISION         ██"
6 PRINT "████████████████████████████████"
7 PRINT " WINNINGS        █"
8 PRINT "████████████████████████████████"
9 PRINT "█      █      █    █      █"
10 PRINT "█      █      █    █      █"
11 PRINT "█      █      █    █      █"
12 PRINT "█      █      █    █      █"
13 PRINT "█      █      █    █      █"
14 PRINT "█      █      █    █      █"
15 PRINT "█      █      █    █      █"
16 PRINT "█      █      █    █      █"
17 PRINT "████████████████████████████████
```

```
████████ "
18 PRINT " ████THE ZX81 ONE ARMED BANDIT█ "
19 PRINT " ███████████████
████████ "
20 PRINT " █
                                       █"
21 PRINT " ███████████████
████████ "
22 PRINT " █  NO BETS ABOVE £100000 █"
23 LET A=7
24 LET P=0
25 PRINT AT 18,4;"    *PLACE YOUR BETS*  "
27 PRINT AT 1,9;"       "
30 INPUT M
34 IF M>100000 THEN GOTO 30
35 IF M=0 THEN GOTO 270
40 PRINT AT 1,9;"£";M
45 LET M=INT (M)
50 FOR N=1 TO 6
55 LET W=INT (RND*6)+1
60 LET X=INT (RND*6)+1
65 LET Y=INT (RND*6)+1
70 LET Z=INT (RND*6)+1
75 LET B=2
80 GOSUB (W*100)+1000
85 LET B=9
90 GOSUB (X*100)+1000
95 LET B=16
100 GOSUB (Y*100)+1000
105 LET B=23
110 GOSUB (Z*100)+1000
120 NEXT N
150 PAUSE 125
160 IF W=X OR Y=Z OR X=Y OR W=Z OR W=Y OR X=Z THEN GOSUB 2500
170 IF W=X AND X=Y AND Y=Z THEN GOTO 350
180 IF W=X AND X=Y AND Y<>Z THEN GOTO 449
190 IF X=Y AND Y=Z AND W<>X THEN GOTO 450
200 IF W=X THEN GOTO 549
210 IF Y=Z THEN GOTO 550
215 FOR N=1 TO 12
217 PRINT AT 3,14;"            "
220 PRINT AT 3,14;"  YOU LOST  "
225 NEXT N
230 LET P=P-M
235 PRINT AT 5,14;"£";P
237 PRINT AT 3,14;"            "
238 IF P=0 THEN PRINT AT 5,14;"£";P;"          "
239 IF P<1000 THEN PRINT AT 5,14;"£";P;"      "
240 IF P<10000 THEN PRINT AT 5,14;"£";P;"     "
241 IF P<100000 THEN PRINT AT 5,14;"£";P;"    "
242 IF P<1000000 THEN PRINT AT 5,14;"£";P;"   "
243 IF P<10000000 THEN PRINT AT 5,14;"£";P;"  "
244 IF P>=20000000 THEN GOTO 4250
250 GOTO 25
270 IF P<0 THEN GOTO 4050
271 IF P=0 THEN GOTO 4150
272 IF P>0 THEN GOTO 4250
349 GOTO 1000
350 IF W=1 THEN GOTO 401
355 FOR N=1 TO 12
356 PRINT AT 3,15;"TOP WINNER"
357 PRINT AT 3,15;"          "
358 NEXT N
360 LET P=(((30*M)+M)+P)
370 GOTO 235
401 FOR N=1 TO 6
402 PRINT AT 7,2;"      ▄     ▖                "
403 PRINT AT 8,2;"      █▄▄    ▚ ▖    ▀█▀ ▄    "
404 PRINT AT 9,2;"      ▄███▄▄   ▄██▌ █   █    "
405 PRINT AT 10,2;"     ██▙██▖ ▖▄█████ █   "
406 PRINT AT 11,2;"     ▀▀███ ▌ ▌▛█▀▀██▀ █   "
407 PRINT AT 12,2;"     ▖ ▀▀██▌ ▛▌█ ▀▀   ▄    "
408 PRINT AT 13,2;"     ▚▘    ▀▀▌ ▌▚▀   ▗▙  ▀  "
409 PRINT AT 14,2;"             ▘ ▘▀    ▝      "
410 PRINT AT 7,2;"▖     ▖█▖ ▖ █▖   ▖█▖   ▖"
411 PRINT AT 8,2;" L     █L     █L     █L   "
412 PRINT AT 9,2;" U▀▜   █U▀▜  █U▀▜   █U▀▜ "
413 PRINT AT 10,2;" C ▌  █C ▌  █C ▌  █C ▌ "
414 PRINT AT 11,2;" K ▌  █K ▌  █K ▌  █K ▌ "
415 PRINT AT 12,2;" Y ▌  █Y ▌  █Y ▌  █Y ▌ "
416 PRINT AT 13,2;"*  * █*  * █*  * █*  * "
417 PRINT AT 14,2;"▘   ▘█▘  ▘█▘  ▘█▘  ▘"
418 NEXT N
420 LET P=(((100*M)+M)+P)
422 GOTO 235
```

```
 449 IF W=1 THEN GOTO 500
 450 IF X=1 THEN GOTO 500
 451 FOR N=1 TO 12
 452 PRINT AT 3,17;"BULLSEYE"
 453 PRINT AT 3,17;"        "
 454 NEXT N
 455 LET P=(((10*M)+M)+P)
 460 GOTO 235
 500 FOR N=1 TO 12
 501 PRINT AT 3,16;"TREBLE 7"
 502 PRINT AT 3,16;"        "
 503 NEXT N
 504 LET P=(((20*M)+M)+P)
 505 GOTO 235
 549 IF W=1 AND W=X THEN GOTO 60
0
 550 IF Y=1 AND Y=Z THEN GOTO 60
0
 551 FOR N=1 TO 12
 552 PRINT AT 3,17;"WINNER"
 553 PRINT AT 3,17;"      "
 554 NEXT N
 555 LET P=(((2*M)+M)+P)
 560 GOTO 235
 600 FOR N=1 TO 12
 601 PRINT AT 3,16;"DOUBLE 7"
 602 PRINT AT 3,16;"        "
 603 NEXT N
 604 LET P=(((7*M)+M)+P)
 605 GOTO 235
1100 PRINT AT A,B;"▖    ▖";AT A+
1,B;" L    ";AT A+2,B;" U▀▀█ ";A
T A+3,B;" C  █ ";AT A+4,B;" K █
 ";AT A+5,B;" Y █  ";AT A+6,B;"
*  * ";AT A+7,B;"▘    ▘"
1150 RETURN
1200 PRINT AT A,B;" ▄▄ ▄▄  ";AT A+
1,B;"█████  ";AT A+2,B;"▀███▀  ";A
T A+3,B;" ▀█▀   ";AT A+4,B;"  ▄█▄█
▖";AT A+5,B;"   █████";AT A+6,B;"
 ▀█▀ ";AT A+7,B;"   ▘  "
1250 RETURN
1300 PRINT AT A,B;"        ";AT A+
1,B;" ▄██▄ ";AT A+2,B;" ▄████▄ ";A
T A+3,B;"██████";AT A+4,B;"████+
█";AT A+5,B;" █████▀";AT A+6,B;"
▀▀▀▀  ";AT A+7,B;"      "
1350 RETURN
1400 PRINT AT A,B;"▖    ▖";AT A+
1,B;"       ";AT A+2,B;" ▄██▄  ";A
T A+3,B;"▄█████▄";AT A+4,B;"▀█████
▘";AT A+5,B;" ▀██▀  ";AT A+6,B;"
  ";AT A+7,B;"▘    ▘"
1450 RETURN
1500 PRINT AT A,B;"  ▄▄   ";AT A+
1,B;"▄▄▄▄▄▄";AT A+2,B;"█▄▄▄▄█";A
T A+3,B;"██▄▄██";AT A+4,B;"██▀▀█
█";AT A+5,B;"█▀▀▀▀█";AT A+6,B;"▘
▀▀▀▀▀";AT A+7,B;"  ▀▀  "
```

```
1550 RETURN
1800 PRINT AT A,B;" ▄▄▄▄   ";AT A+
1,B;"  ▞    ";AT A+2,B;" ▞  *  ";A
T A+3,B;"█▄▄▄  ";AT A+4,B;"    █
█";AT A+5,B;" *  ▚▞ ";AT A+6,B;"
 ▞▚ ";AT A+7,B;"  ▘ ▘▘"
1850 RETURN
2500 PRINT AT 18,4;"NUDGEANUDGEB
NUDGECNUDGED"
2505 INPUT J$
2510 IF J$="A" THEN GOSUB 3000
2515 IF J$="B" THEN GOSUB 3050
2520 IF J$="C" THEN GOSUB 3100
2525 IF J$="D" THEN GOSUB 3150
2530 RETURN
3000 LET B=2
3005 FOR N=1 TO 6
3010 LET W=INT (RND*6)+1
3020 GOSUB (100*W)+1000
3030 NEXT N
3040 RETURN
3050 LET B=9
3055 FOR N=1 TO 6
3060 LET X=INT (RND*6)+1
3070 GOSUB (100*X)+1000
3080 NEXT N
3090 RETURN
3100 LET B=16
3105 FOR N=1 TO 6
3110 LET Y=INT (RND*6)+1
3120 GOSUB (100*Y)+1000
3130 NEXT N
3140 RETURN
3150 LET B=23
3155 FOR N=1 TO 6
3160 LET Z=INT (RND*6)+1
3170 GOSUB (100*Z)+1000
3180 NEXT N
3190 RETURN
4000 SAVE "B"
4001 GOTO 4005
4005 PRINT AT 6,5;"*MONEY*MONEY*
MONEY*MON"
4010 PRINT AT 7,5;"M▀▀▀▀▀▀▀▀▀▀▀▀▀▀
▀▀▀▀▀▀▀▀▀E"
4011 PRINT AT 8,5;"O█ ▛▖▛▜▛▖▛
       Y"
4012 PRINT AT 9,5;"N█ ██▙▟▛▛
GAMBLE █*"
4013 PRINT AT 10,5;"E█ ▙▘████▙▄
   ON    M"
4014 PRINT AT 11,5;"Y█  ▖▖▄▄▖▖▛
  YOUR    D"
4015 PRINT AT 12,5;"*█  ▙████ ▟
  ZX81    N"
4016 PRINT AT 13,5;"M█   ████ ▘
         E"
4017 PRINT AT 14,5;"O█   ▘▀▀▀▀ ▘
         Y"
```

```
4018 PRINT AT 15,5;"N▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄*"
4019 PRINT AT 16,5;"EY*MONEY*MONEY*MONEY**"
4020 PRINT AT 18,5;" PRESS NL TO START "
4025 INPUT G$
4030 IF G$<>"" THEN GOTO 4020
4035 IF G$="" THEN GOTO 1
4050 CLS
4055 PRINT AT 0,0;"████████████████████████████████"
4060 PRINT AT 1,0;"███";CHR$ 151;"█████";CHR$ 151;"███████";CHR$ 151;"███";CHR$ 151;"█████"
4065 PRINT AT 2,0;"█";CHR$ 151;"█████";CHR$ 151;"███████";CHR$ 151;"█████████";CHR$ 151;"██"
4070 PRINT AT 3,0;"████████████████████████████████"
4075 PRINT AT 4,0;"████";CHR$ 151;"██████";CHR$ 151;"████████████";CHR$ 151;"████"
4080 PRINT AT 5,0;"███";CHR$ 151;"████████████████████████"
4085 PRINT AT 6,0;"████████████";CHR$ 151;"██";CHR$ 151;"████████";CHR$ 151;"██";CHR$ 151;"██"
4086 PRINT AT 7,0;"████ RIP ▀█";CHR$ 151;"██████";CHR$ 151;"████";CHR$ 151;"██████";CHR$ 151
4087 PRINT AT 8,0;"████ ▀▀ ▀█ ██";CHR$ 151;"█████████";CHR$ 151;"█";CHR$ 151;"███"
4088 PRINT AT 9,0;"████ ALL █ ████████";CHR$ 151;"███████";CHR$ 151;"█"
4089 PRINT AT 10,0;"███ PERSONS ██████";CHR$ 151;"██████████"
4090 PRINT AT 11,0;"███ WHO ████";CHR$ 151;"██████████████"
4091 PRINT AT 12,0;"███ NEVER ████████████████████"
4092 PRINT AT 13,0;"███ PAYED ████████████";CHR$ 151;"██████"
4093 PRINT AT 14,0;"███████████████████████████"
4094 PRINT AT 15,0;"███████████████████████████"
4095 PRINT AT 16,0;" ▀██████ ▙ "
4096 PRINT AT 17,0;" ▀██████ ▙▄ "
4097 PRINT AT 18,0;" ▀██████ ▀ PRESS N/L TO "
4098 PRINT AT 19,0;" ▀▀ ▀ GET EVEN "
4099 PRINT AT 14,5;P;
4100 INPUT L$
4101 IF L$="" THEN GOTO 1
4102 IF L$<>"" THEN GOTO 4100
4150 CLS
4151 PRINT AT 5,5;" ▛  ▌ ▖ ▌ ▌▌ ▜"
4152 PRINT AT 6,5;" ▌ ▛▛.▛▌▞▌▜▛▌▌ ▌"
4153 PRINT AT 7,5;" ▌ ▌ ▌▌▌▌ ▛▖▌▀▌▌▌▌ ▌"
4154 PRINT AT 8,5;" ▌ ▙▌▌▌▌▙▌▚▌▟▌▌▖▖ ▌"
4155 PRINT AT 9,5;" ▌ --------------- ▌"
4156 PRINT AT 10,5;" ▌ ONLY BREAK EVEN ▌"
4157 PRINT AT 11,5;" ▌ --------------- ▌"
4158 PRINT AT 12,5;" ▌  PRESS NL  FOR  ▌"
4159 PRINT AT 13,5;" ▙  FRESH  GAME  ▟"
4160 INPUT E$
4165 IF E$<>"" THEN GOTO 4160
4170 IF E$="" THEN GOTO 1
4250 CLS
4251 PRINT AT 5,2;"▛▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▀▜"
4252 PRINT AT 6,2;"▌BANK OF ZX81 ▐"
4253 PRINT AT 7,2;"▌------------ ▐"
4254 PRINT AT 8,2;"▌ ▐"
4255 PRINT AT 9,2;"▌PAY THE BEARER: ▐"
4256 PRINT AT 10,2;"▌WHO HIS VERY LUCKY, BECAUSE ▐"
4257 PRINT AT 11,2;"▌ ANYMORE WOULD BE TOO MUCH ▐"
4258 PRINT AT 12,2;"▌  FOR THE BANK MANAGERS ▐"
4259 PRINT AT 13,2;"▌  NERVOUS SYSTEM TO STAND ▐"
4260 PRINT AT 14,2;"▌ ▐"
4265 PRINT AT 15,2;"▌ PRESS NL  FOR ME TO EVEN UP ▐"
4270 PRINT AT 16,2;"▙▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▄▟"
4274 PRINT AT 9,18;"£";P;
4275 INPUT Y$
4280 IF Y$<>"" THEN GOTO 4275
4285 IF Y$="" THEN GOTO 1
```

The Fuller phoenix

In the dim, dark ages Fuller gained a good reputation as suppliers of good add on units, sound boxes, keyboards etc. for the ZX81.

With the production of the Spectrum they went for a redesigned up market replacement keyboard and from then on they never seemed to be out of trouble.

A friend of mine was attempting to get one and the range of "reasons" for non delivery was varied indeed: The quality of the first batch was not acceptable so we're behind with our orders. There has been a delay at the factory. We've redesigned it to allow for the Interface I and Microdrives, We've had a fire on the premises. . .

A great pity, and when the inevitable liquidation of the company happened a lot of people had their fingers burnt.

So Nordic Keyboards of Merseyside took a big chance when they decided to retain the Fuller name when they took over the manufacturing and distribution of the companies products.

The FDS Executive is the updated version of the keyboard and looks identical except that the board is connected by a ribbon cable to the user port and does not need the top of the Spectrum to be removed. The price is also slightly higher at £59.95.

Personally I wish them well and I look forward to reviewing this and other products they produce, although I can understand any reticence on the buyers side to purchase mail order, it is worth looking at them in your local store.



Opus option



Opus supplies Ltd. have made their mark as a supplier of computer hardware. The first item to come to my notice was their budget priced monitor from JVC advertised on the computer press. I have been supplied with one and will report elsewhere how it performs when used in conjunction with the Adapt Electronics RGB output unit for the Spectrum.

The latest addition to their monitor range is the high resolution version available for the QL at £299.95, it will also work with the Spectrum and most other popular computers.

Disk drives are another area in which Opus are lowering prices, they have introduced a £5.25 single sided, 100K drive for £119.95, however if you intend to use it with a Spectrum via the excellent Technology Research interface then you will have to get a power supply fitted and this will cost you extra.

A 3 inch drive is also available under their brand name, this is a double sided drive giving 250K (500K double density) and costs £119.95. No information as to whether a PSU is fitted was given. Of course with all this hardware around you'll need something to put it on, and Opus have ventured into this area as well.

They have a range of computer furniture for both the business and the home markets. Prices range from £60.00 for the home unit to £240.00 for the best business desk.

And now, if you don't live in London you are still able to avail yourself of their goods as Boots have agreed to stock their products in their chain of stores.

In . . . . Brief

● Stephen Adams has several items of interest on the market, the PZ1 tape controller for the ZX81 Spectrum will control up to two cassette decks via the remote socket.

A series of adaptors for using many of the ZX81 peripherals on the Spectrum are also available for around £9.00. Further information can be obtained from him at 1 Lesiwn Rd. London N16 7NL.

● From AGF comes an invaluable accessory, a small extender which plugs into the back of your ZX81 or Spectrum with an identical port on the other end, this has the effect of moving the port about half an inch out from the computer.

This is perfect if you have a non standard keyboard fitted and want to use one of the tight fitting "lipped" interfaces. It is also useful if you are frequently changing peripherals as it is cheaper to replace if the extender becomes worn than the computer.

These simple but precious devices cost £1.99 each and there are three versions available, a ZX81, Spectrum and an angled one which is designed to bring the Currah speech unit back over the top of any other units so that the rather short leads fitted to it reach the sockets without strain.

● Cambridge Systems Technology are developing their QL interfaces, their Q-PI Centronics parallel interface has been reduced to £50.00 + VAT.

They also have a full implementation of the international IEEE-488 protocol interface on sale, the Q-488. Its cost was not available at the time of going to press, but more information can be obtained from CST, 30 Regent St., Cambridge, CB2 1DB.

● A stylish computer trolley is available from Paul Gallini Enterprises, 146 Myton Rd., Warwick, CV34 6PR for £65.00 inc.

● A super looking four colour printer/plotter is now available from Datafax Ltd., Basingstoke, Hants for the unbelievably low price of £199.95.

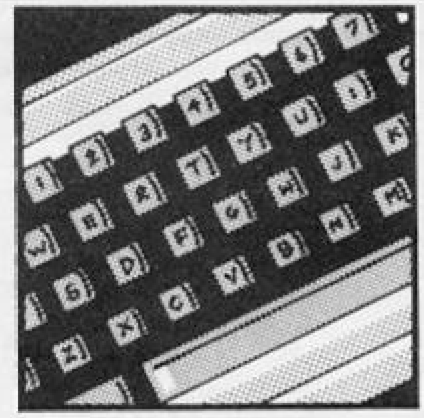
I hope to be looking at one end and reporting back on it in the near future.

● Prices reach a new low with Fischertechnics new six-in-one robot arm. This is supplied in kit form and contains all the electronics needed as long as an "appropriate" interface is used on the controlling micro.
The price is £64.95!!



First steps in Machine Code

Part 2: Working with registers



An introduction to Z80 Machine Code by David Nowotnik

Last time I introduced to you the concepts of the Z80 microprocessor, machine code and assembly language programming. Using those concepts, I gave a few simple examples of moving data to and from the Z80 CPU. To remind you of the internal layout of this CPU, the diagram of the CPU is shown again this time (fig. 1). as I'll be referring to it again as I develop some more of the ways in which machine code can transfer data from memory to the internal registers of the CPU. But before we begin with these aspects of the machine code, I'd like to introduce you to a different way to deal with numbers when working with computers. This numeric system is called hexadecimal (commonly abbreviated to hex).

Hexadecimal

The numeric system we use in every day life is called decimal. To use decimal, we have 10 characters (0 to 9) which we combine to define a number. Decimal is easy for us to learn as we have ten fingers to use in counting.

The computer, however, finds decimal cumbersome as it has eight 'fingers'; the number of bits in a register or a bytes in memory. When using BASIC, the interpreter in ROM translates numbers to decimal to make life easier for us. But in using machine code, it becomes easier to adopt the computer's numeric system. This is a system which has 16 characters ('two hands' of eight 'fingers') which make up all numbers, and is called hexadecimal.

Hexadecimal works like this: The numbers 0 to 9 in decimal are the same in hexadecimal (or hex, for short). 10 decimal becomes 'A' in hex, eleven is 'B', and so on up to 15, which is 'F' hex. Sixteen is then '10' hex, and 26 decimal is '1A' hex. The highest value in hex with just two characters is 'FF', which is 255 in decimal. And that just happens to be the highest value that can be stored in a byte (or in a single register). So, every byte value can be defined by a two character number using hex (e.g. 0A for ten, FA for 254 decimal).

Some of the other advantages of hex will become apparent as we go through the series. To encourage you to use hex rather than decimal, I have incorporated a hex to decimal conversion program (suitable for both ZX81 and Spectrum), in Fig.2, and, later, two new machine code loaders which use hex rather than decimal, as was used last time. But, for now, back to machine code.

More LOADing

I introduced last time the assembly language command of LOAD. It happens to be the most frequently used machine code operation of all. There are several variations on LOAD; the simpler and most frequently used will now be described.

The simplest of all are the series of LOAD commands which allow the contents of one register to be copied into another register. The BASIC equivalent is LET B=C (for LD B,C); the originating register remains unaltered, but the value it holds (a number between 00 and FF hex) is copied into the other register. A single opcode carries out this operation; table 1 displays the opcodes for all possible inter-register transfers. There is no equivalent instruction to copy the contents of one register pair into another pair, but this can be achieved by two single register transfers, e.g.

```
LD      H,B
LD      L,C
```

to copy BC into HL.

It is, of course, necessary to load values into the registers from outside the CPU. One way to do this is to load data from the program. For example, the instruction:

```
LD      B,n
```

will load the B register with a value represented by 'n'. In machine code, LD B,n appears as a two byte instruction. The first byte is 06 hex, which is the opcode for LD B,n. The value in the memory address immediately following this opcode is the value to be loaded into the B register. This is known as the operand. For example, if the two bytes (opcode and operand) were 06FB, then the value FB would be loaded into the B register. All the other registers can be loaded from the program in the same way, and all the opcodes for these instructions appear in table 2. Remember, you must have a second byte to complete this instruction.

You can, if you wish, load a register pair directly with a single instruction, rather than use two instructions. From last time, you will recall that the H and L, B and C, and D and E registers can be paired such that they effectively can hold any number between 0 and 65535 (00 to FFFF hex). C,E, and L form the low byte of the pair, while B,D, and H are the high bytes.

Three bytes make up the machine code instruction to load a register pair directly. The first is the opcode, the second is the value which goes in the low byte, and the third the value to go in the high byte. Note that the low byte preceeds the high byte in a two byte number. This is a Z80 conversion, and we'll see more of that in a moment. The two byte load instructions also appear in Table 2.

The direct load instructions are equivalent to, for example, LET B=5, or LET BC=1225 in BASIC. For more flexibility, you may wish to load the equivalent of a variable (i.e. the contents of an address in memory) into one of the registers. There are a number of ways you can do this.

For reasons which will become apparent later, the A register is special. For example, it is the register in which all one byte arithmetic is carried out. So, there are more options to LOAD A than any other single register. You can load the A register with the contents of a byte from a specific memory address with LD A,(nn). 'nn' represents a memory address, and you may remember from last time that the brackets mean 'the contents of'. LD A,(mm) is a 3 byte instruction in machine code; the fist byte (3A) is the opcode, the second and third are the low and high byte respectively of the address of the byte whose value is loaded into A.

Similarly a register pair can be loaded directly from memory; i.e. LD dd, (nn), where dd represents HL, DE, or BC. When dd is HL, there is a single opcode (2A), followed by a two byte operand. To load BC or DE a two byte opcode is required, followed by the address operand. The address operand points to the byte whose value is loaded into the low register of the pair; the next address in memory after the operand address is loaded into the high register of the pair.

For yet more flexibility, you can use the HL register pair to point to an address in memory whose value is loaded into a register. Thus, LD r,(HL) — where r represents any register — is an instruction requiring only one byte (the opcode) which takes the number stored in the HL register as the address in memory from which a value is copied into the register. The A register allows the BC and DE register pairs to perform as pointers as well (e.g. LD A,(BC)).

Whenever it is possible to ▶

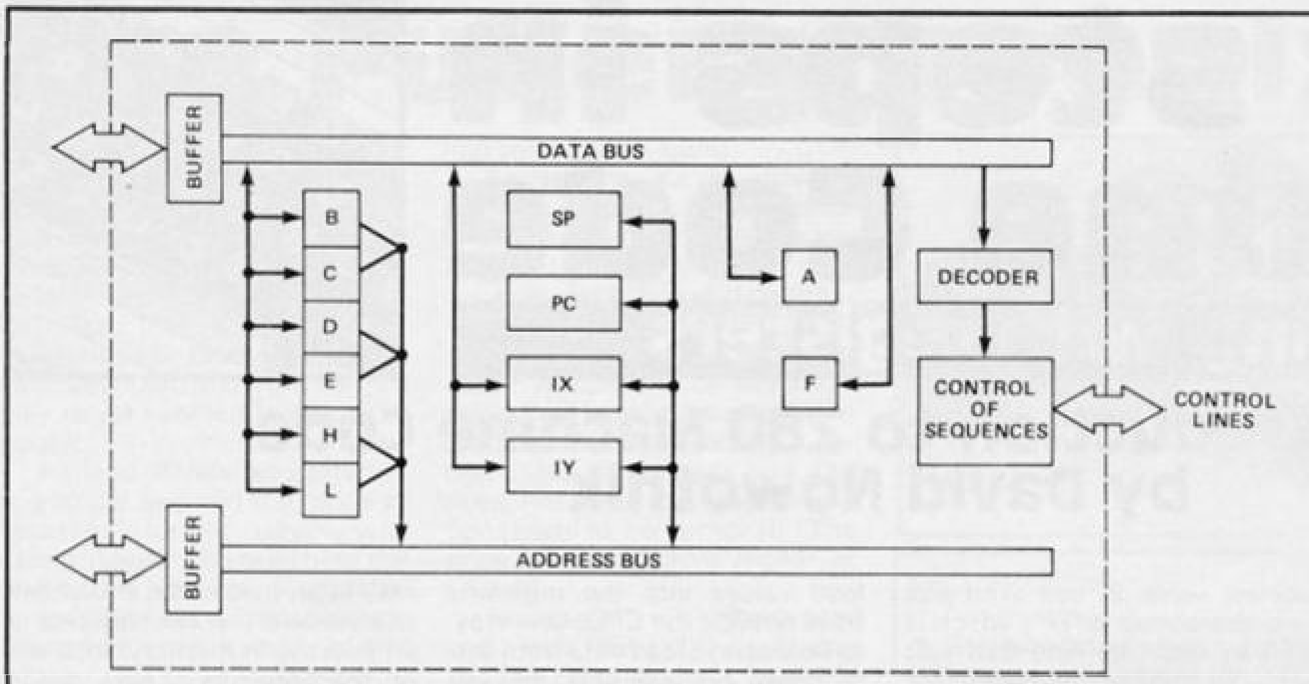


Figure 1. Internal Architecture of the Z80 CPU

load a register (or register pair) from an address in memory, it is also possible to copy the register value into memory. For example, LD (HL),A will copy the contents of the A register to an address in memory indicated by the value of the HL register.

The final LOAD instruction for now allows you to place a value into RAM without going through one of the registers of the CPU. That is LD (HL),n. A two byte instruction, the second being the value which goes into the address indicated by the value held in the HL register pair.

Well, that was a lot of theory in a short space. If it wasn't clear, read the section again, and look at Tables 2 and 3, which contain opcodes for all the LOAD instructions dealt with just now. Several of them will appear in the examples I give later on, which should also help to clarify these instructions.

Simple Arithmetic

Computers wouldn't get very far if all you could do was to transfer bytes of data from one place to another. Much of the rest of the series will deal with how registers can be manipulated; we'll start now with some simple arithmetic. Next issue I'll demonstrate addition and subtraction in machine code, but if all you want to do is add or subtract the number one from a register (or register pair) then the Z80 provides a simple way of doing it.

The instructions INC and DEC will, respectively, add or subtract one from any register or register pair. All these instructions are just one byte long (the opcode), and their values are shown in table 4. As an example, if the value of the A register is 3A, then INC A will increase it to 3B, and DEC A will decrease it to 39 hex. If the value of A were FF (the highest possible) then INC A would convert the value in A to zero; similarly, DEC A when A is zero would make A hold the value FF.

On paired registers, INC and DEC work on the combined value held by the pair of registers. So, beware, INC HL is not the same as INC H and INC L.

Again, the examples should clarify any doubts about INC and DEC.

A Simple Loop

With the machine code I have given you so far, it would only be possible to go to the start of a code (the USR function), carry out a list of instructions, and, with the RET instruction, return to BASIC. From BASIC, you will have learnt the power and utility of FOR . . . NEXT loops; it's possible to do a similar thing in machine code. The simplest way is with a complex instruction DJNZ n.

The letters stand for Decrement B and Jump if Not Zero. What it does is to use the B register as a counter, just like the variable in a FOR . . NEXT loop. When this instruction is encountered, the B register is decremented, and its value is tested. If it is not equal to zero, then the Program Counter (remember that from last time?) will jump to a value governed by the value of n, the operand to DJNZ (this is a 2 byte instruction). If the value of B is zero, then PC is incremented in the normal way, such that the next instruction (immediately following the DJNZ instruction) is implemented.

How the value of n controls the jump is quite complicated. It allows you to jump both forward and backwards; this is called relative jumping (as PC is altered relative to its current position). Here's how it works;

If the value of n is between 0 and 127, then the program counter jumps forward by the value of n; i.e. PC = PC + n. But if the value of n lies between 128 and 255, then the program counter jumps back according to the sum PC = PC − (256-n). The most commonly occuring fault in writing maching code is the miscalculation of a relative jump, and there is little wonder why! One of the great benefits of writing in assembly language, and having an assembler program translate to machine code is that the calculation is carried out for you. If you do calculate relative jumps for yourself, then remember that the starting point for the sum is the address of the opcode immediately following the DJNZ instruction. That's the place you would end up if you had DJNZ 0.

For simple loops in machine code, you will be jumping back following the DJNZ instruction, and you'll see that in the examples I will be giving. You'll also notice that every time a loop with DJNZ is set up, the B register is filled with an appropriate value, much like the FOR statement in a FOR . . . NEXT loop.

One final opcode for now; you'll see it in some of the examples — that is NOP. It simply means do nothing! It is machine codes way of carrying out PAUSE. NOP only slows down machine code for a fraction of a second, so you'll need quite a few NOP's (in a loop) to see any effect.

Phew! That was a lot of theory in a small space. Let's try some examples now as light relief, and hopefully, to clarify the theory. You'll notice that I have split up examples for the ZX81 and Spectrum. That's because the differences between the machines are quite considerable when it comes to machine code. So, GOTO the

Figure 2. Hex to Dec conversion program.

```
 10 LET A$ ="0123456789ABCDEF"
100 INPUT B$
110 IF B$(1)="$" THEN GOTO 300
120 LET Z=VAL B$
130 LET A=Z
140 LET H$ =""
150 IF A=0 THEN GOTO 400
160 LET R=INT (A/16)
170 LET S=A-16*R
180 LET H$ =A$(S+1)+H$
190 LET A=R
200 GOTO 150
300 LET H$ =B$(2 TO)
310 LET Z=0
320 FOR J=1 TO LEN H$
330 FOR I=1 TO 16
340 IF H$(J)=A$(I) THEN GOTO 370
350 NEXT I
360 STOP
370 LET Z=Z+(I-1)*16**(LEN H$ -J)
380 NEXT J
400 PRINT Z,H$
410 GOTO 100
```

This program interconverts decimal and hex numbers. When you run the program, enter a decimal number or a hex number with a '$' sign prefix. This tells the program (line 110) whether the number is hex or decimal. Both Hex and decimal numbers will be printed on the screen.
*N.B. In line 370, the double star '**' is to the power of' on the ZX81. On the Spectrum, this should be '∧'.*

Table 1. Opcodes for loading between registers.

LD r,r'

Load from: (r)

		A	B	C	D	E	H	L
Load to:	A	7F	78	79	7A	7B	7C	7D
	A	47	40	41	42	43	44	45
(r')	C	4F	48	49	4A	4B	4C	4D
	D	57	50	51	52	53	54	55
	E	4F	58	59	5A	5B	5C	5D
	H	67	60	61	62	63	64	65
	L	6F	68	69	6A	6B	6C	6D

Table 2. Load registers with numbers from the program.

LD r,n		LD dd,nn	
LD A,n	3E	LD BC,nn	01
LD B,n	06	LD DE,nn	11
LD C,n	0E	LD HL,nn	21
LD D,n	16		
LD E,n	1E		
LD H,n	26		
LD L,n	2E		

Table 3. Loading registers to and from memory.

Loading from memory		Loading to memory	
LD A,(nn)	3A	LD (nn),A	32
LD A,(BC)	0A	LD (BC),A	02
LD A,(DE)	1A	LD (DE),A	12
LD A,(HL)	7E	LD (HL),A	77
LD B,(HL)	46	LD (HL),B	70
LD C,(HL)	4E	LD (HL),C	71
LD D,(HL)	56	LD (HL),D	72
LD E,(HL)	5E	LD (HL),E	73
LD H,(HL)	66	LD (HL),H	74
LD L,(HL)	6E	LD (HL),L	75
LD BC,(nn)	ED4B	LD (nn),BC	ED43
LD DE,(nn)	ED5B	LD (nn),DE	ED53
LD HL,(nn)	2A	LD(nn),HL	22

(LD (HL),n 36)

Table 4. Increment and Decrement registers.

	A	B	C	D	E	H	L	BC	DE	HL
INC	3C	04	0C	14	1C	24	2C	03	13	23
DEC	3D	05	0D	15	1D	25	2D	0B	1B	2B

Figure 3. Hex loader program.

```
10 POKE 16388,47
20 POKE 16389,117
30 LET M$ = "  "
40 LET X = 30000
50 FOR C = 1 TO LEN M$ STEP 2
60 POKE X, (16 * (CODE M$ (C) - 28) + (CODE M$ (C + 1) - 28))
80 NEXT C
```

Example 1.

```
30 LET M$ = "2A0C40062023362110FBC9"
```

Example 2.

```
30 LET M$ = "2A0C403E161194751206202336 2110
FB231A473D1210F2C9"
```

Example 3.

```
30 LET M$ = "2A0C40119475 3E1E1223362106
FF0010FD36201A473D1210F0C9"
```

Figure 4. Assembly language listings for ZX81. Example 1.

```
      LD HL,(16396)  2A0C40   Start of display file
      LD B,32        0620     loop counter set
LOOP  INC HL         23       Next place in display
      LD (HL),33     3621     Put '5' on the screen
      DJNZ LOOP      10FB     Jump back to LOOP
      RET            C9       or return if 32 done.
```

Example 2.

```
      LD HL,(16396)  2A0C40   Start of display file
      LD A,22        3E16     Number of rows
      LD DE,30100    119475   Place to store row
      LD (DE),A      12         counter
ONE   LD B,32        0620     No. of columns
TWO   INC HL         23       Next screen position
      LD (HL),33     3621     Put '5' on screen
      DJNZ TWO       10FB     Jump back to 'TWO'
      INC HL         23       Jump over NEWLINE
      LD A,(DE)      1A       Check on row counter
      LD B,A         47
      DEC A          3D
      LD (DE),A      12       Store the row counter
      DJNZ ONE       10F2     Jump back to 'ONE'
      RET            C9       unless 22 rows complete
```

Example 3.

```
      LD HL,(16396)  2A0C40   Start of display file
      LD DE,30100    119475   Column counter
      LD A,30        3E1E     Number of columns
      LD (DE),A      12       stored
ONE   INC HL         23       Next screen address
      LD (HL),33     3621     Place '5' on the screen
      LD B,255       06FF     Delay counter
TWO   NOP            00       Delay loop
      DJNZ TWO       10FD
      LD (HL),0      3600     Erase '5'
      LD A,(DE)      1A       Check on column counter
      LD B,A         47
      DEC A          3D
      LD (DEC),A     12       Replace counter value
      DJNZ ONE       10F0     Jump back to ONE
      RET            C9       Unless all done.
```

Figure 5. Spectrum Hex loader.

```
10 CLEAR 29999
20 LET X = 30000
30 READ a$ : IF a$ = "s" THEN STOP
40 FOR i = 1 TO LEN a$ STEP 2
50 LET y = 0
60 FOR j = 0 TO 1
70 LET r = CODE (aS(i + j))
80 LET s = r-48-(r  57) * 7
90 LET y = y + 16 * s * (j = 0) + s * (j = 1)
100 NEXT j
110 POKE 'x,y: LET x = x + 1
120 NEXT i: GOTO 30

2000 DATA "s"
```

Example 1.

```
1000 DATA "2100581194753E1612"
1010 DATA "062036872310FB1A473D"
1020 DATA "1210F3C9"
```

Example 2.

As above, but change line 1000 to

```
1000 DATA "210040119475 3EC812"
```

Example 3.

```
1000 DATA "2100581194753E0612"
1010 DATA "3E003295753A95750680"
1020 DATA "772310FC06083C10FD"
1030 DATA "3295751A473D1210E9C9"
```

appropriate section!

ZX81 Examples

To try the machine code examples in this issue you'll need the hex loader in fig. 3. Type this in and SAVE it. The machine code goes in as a string of hex values in line 30, so substitute the line 30 in examples one to three, then RUN, followed by the direct command RAND USR 30000 to activate the routines.

All examples use the display file; that is, you'll see the effect of the machine code on the screen.

In example 5, you'll get a row of '5's printed on the top line of the screen. In example two, you'll get a screen full of '5's — instantly (try that in BASIC). The final example will give you a '5' dashing across — don't blink or you'll miss it.

Space doesn't permit too full an explanation of how these routines work. Try them out for yourself, look at the assembly language listing, and see if you can work out what is happening. You'll get some more clues next issue.

Also, try varying the hex numbers underlined (generally use a smaller number) and see what happens.

Spectrum Examples

My Spectrum machine code hex ▶

loader appears in Fig. 5. First enter it, then save it. The machine code appears in DATA statements in line 1000 onwards, so for each example, type in the DATA lines appropriate for that example. In all cases leave line 2000 as the last line; this contains a symbol ('s') that informs the program you have come to the end.

From the details given in the assembly language listing you should be able to work out what is happening. all examples use the Spectrum's display or attribute file, so the effect is always visual.

With the appropriate DATA lines in place (in fig 5) RUN the program, then call the machine code with RANDOMIZE UR 30000, and see what happens. Try experimenting by changing those numbers which are underlined in Fig 5. Good luck!

Figure 6. Assembly language listings for Spectrum. *Example 1.*

```
      LD HL,22528   210058  Start of the attributes
      LD DE,30100   119475  Storage byte
      LD A,22       3E16    Loop counter
ONE   LD (DE),A     12      Stored at 30100
      LD B,32       0620    Column counter
TWO   LD (HL),127   367F    Attribute value
      INC HL        23        placed in row
      DJNZ TWO      10FB
      LD A,(DE)     1A      Recover loop counter
      LD B,A        47      copy in B
      DEC A         3D      Loop counter minus one
      LD (DE),A     12      Replace loop counter
      DJNZ ONE      10F3    Jump back to ONE
      RET           C9        unless all done.
```

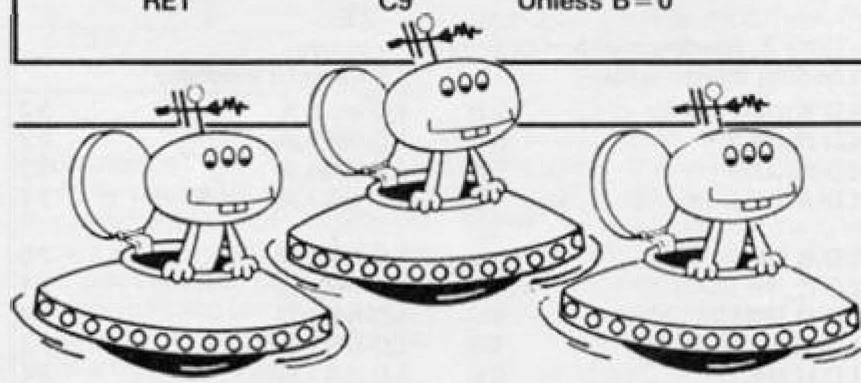
Example 2.

This is the same as 1 except for the first three lines:

```
      LD HL,16384   210040  Start of display file
      LD DE,30100   119475  Storage byte
      LD A,200      3EC8    Loop counter.
```

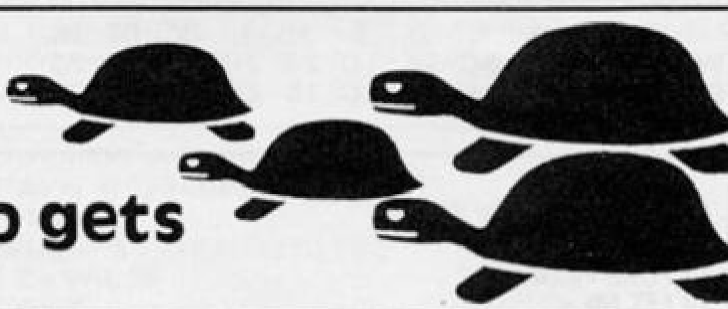
Example 3.

```
      LD HL,22528     210058  Start of attributes
      LD DE,30100     119475  Storage byte
      LD A,6          3E06    First loop counter
      LD (DE),A       12        stored
      LD A,0          3E00    Attribute value
      LD (30101),A    329575    stored
ONE   LD A,(30101)    3A9575  Recover attribute value
      LD B,128        0680    Loop counter
TWO   LD (HL),A       77      Place attribute in file
      INC HL          23        next position
      DJNZ TWO        10FC      until 128 done
      LD B,8          0608    Add eight
TRE   INC A           3C        to attribute value
      DJNZ TRE        10FD
      LD (30101),A    329575  Store new value
      LD A,(DE)       1A      REcover counter
      LD B,A          47      Copy into B
      DEC A           3D      Counter minus one
      LD (DE),A       12      Store new counter value
      DJNZ ONE        10E9    Jump back to ONE
      RET             C9      Unless B=0
```



WHAT?

Tortoise-Wise



or more lines from a parent who gets left behind. By David Stewart

I'm still struggling. But why worry? It's one of the things we parents are supposed to be good at, isn't it?

Remember the tale of the Tortoise and the Hare? Well I'm the Tortoise and I've got at least two hares to contend with. Right now the house is quiet. Both my sons have gone to bed, both their one track minds are probably pursuing flapping lavatory seats through countless mazes, zapping their way through their slumbers, counting the screens until morning.

The Spectrum is cooling down at the end of the day. The joystick is still.

The telephone rings. Thankfully there are still some instruments of modern technology I know how to use. I answer it. It is another Tortoise. His name is Peter and his son too has a Spectrum.

"What can I do for you Pete?"

"Have your kids gone to bed yet?"

"Yes. Why?"

"I'm putting a program in from one of those magazines, thought I might learn something doing it this way. Doesn't mean anything to me of course but then nor does the manual much. Anyway I think there's something wrong somewhere and my kid's gone to bed too."

"Can I help?" I ask.

"I doubt it. I was hoping one of your kids was still up so I could ask him. Never mind. Must go."

We Tortoises should stick together, I think to myself. We shouldn't be asking the Hares for help. There is such a thing as loyalty and safety in numbers.

The Tortoise moves on late into the night. Now is my chance to make up some ground. I settle myself in front of the TV monitor, Spectrum annual next to the keyboard.

It's exciting in a way that the Hares asleep upstairs have long forgotten. It's new to me still. I feel like someone in the Twenty First Century. They do it as naturally as sitting down to fish and chips. I'm still filled with a sense of awe. I'm still impressed. I still don't understand enough. To them, the computer is a tool, a toy, a game. It takes a lot to impress them. Matthew Smith impresses them, screen graphics impress them. Hackers impress them. But I'm still trying to get words like INPUT, POKE, PEEK, strings and vaiables relocated in an already overtaxed vocabulary.

They are the hares and I'm the Tortoise. I know I'm not keeping up. Tortoise-wise, I'm not giving up either. I'll get there in the end. A tortoise I may be. A Dinosaur I'm not. Yet.

I plod on through the manual wondering if being useless at languages has got anything to do with the pitifully slow progress I make. Or is it Maths I was useless at? Or both? And why do they call BASIC a language? And if there are other languages for computers are there any more suited to Tortoises? What are the differences anyway? And if I ask the hares in the morning I know they'll laugh and I'll go back into my shell, Tortoise-wise.

Music Micro Please!



An outline of some of the more harmonious programs and devices recently produced. Ray Edler takes up the Baton and conducts the investigation.

For the professional/dedicated/ expert musician there is the Midi interface and synthesiser system, but for many others, including yours truly, the cost of such equipment is beyond the realms of an overdraft.

For us interested but impoverished would be musicians there are a few more affordable, though much less versatile, alternatives.

Forte!

An essential first step is to do something about the BEEP, there are several ways around this, you can simply amplify the thing or, if you want to get the most from the computer, then you will need to purchase a unit which incorporates a sound chip. Most of these units use a version of the AY-3-8910/ 8912 chip and this is a well tested and reliable chip which can produce some impressive sounds.

Simple amplification of the BEEP can be achieved in several ways and is the least expensive of the two options. Beep amps either have a built in speaker or send the sound to the TV and some such units are:

THE CHEETAH BEEP AMP, this puts the sound through the TV and is in the same size box as their other units, complete with a full through port for other peripherals. Cost £9.95.

CURRAH MICRO SPEECH unit. This is primarily a speech unit but it also routes the Spectrum Beep through the TV and is worth considering if you want to investigate the world of speech as well. Cost £29.95.

DK'TRONICS BEEP AUDIO AMP. This is a self contained unit in two parts, the amplifier interface and a separate 1" pod mounted loudspeaker. The interface has a full through port and, although lipped, fits both versions of the Spectrum securely. Cost £14.95.

THE TRICHORD, a full sound box including AY chip, amp and speaker and software for both ZX81 and Spectrum, available from Newtech Developments Ltd. Cost ranges from £24.95 to £26.95 depending on model and internal amp fitted or not.

WILLIAM STUART SYSTEMS sound generator unit, designed to go with their speech recognition and synthesiser units but can be used on its own (if plugged into a stereo system it sounds superb). Software is available separately and is excellent, the **ARP** program kept me fascinated for hours.

FULLER (reinstated by Nordic) have a sound unit and a master unit which includes speech, we have not tried one out yet, but the new look packaging is nice.

And finally **DK'TRONICS** again, identical to their beep amp in size and shape is a full Sound Synthesiser. This is based on the AY chip and is supplied with a demo tape. Cost £29.95.

The Dk'tronics 3 Channel Sound Unit

I will give a detailed account of this unit as it is one of the few still left in my possession, most of the others had to be returned after each specific review, and anyway its one of the latest on the market, and most are similar in operation.

This is supplied with an eight page booklet which gives setting up details, the volume is adjustable by a small screwdriver through a little hole in the top of the unit, full volume is loud enough for most purposes!

Some demo programs are given which explain the basics of programming the beast, these are in BASIC but a machine code programmer can glean enough information to enable him/her to incorporate routines in their programs.

Most of the manual is dedicated to operating instructions for the program supplied, these are brief and there are gaps which left me confused – the use of the Envelope generator for instance. The software itself is good and allows you to create tunes in 3 part harmony and is cursor controlled, this means that it can be operated by a joystick. However I found it a little tedious to use and the editing is almost non existent, an error early on means re-entry of the rest of the line.

During playback the three lines of music are shown as a letter/number representing note and octave. Without prior musical knowledge you may find it hard to get anything of worth from it.

The sound quality itself is very good, the speaker handles the output with minimal distortion and many a long hour was happily spent with this program.

A unit which is nicely made, looks and sounds good and is well worth considering.

It also works well with other peripherals, the Technology Research Disk drive unit functioned without problems and it was easy to convert the software supplied to disk. The ZXLprint III centronics unit also sat at the back quite happily, and of course DK's own units, joystick interface etc. proved compatible.

Make Music

is a program produced by the **Buffer micro shop** which works in conjunction with any Spectrum sound unit using the AY-3-8912 chip and attempts to emulate a three track recorder, similar to the BBC Music Processor program. It is quite successful although rather awkward to operate, the screen display is not as informative as the BBC, and envelope control is non existent. Good, but not as versatile as it could have been, and as far as I know, the only one of its type for the Spectrum.

XOR from a company called Xorsoft and described by reviewer Clive Smith as "one of the better ones" and Music Maker from Malan are two programs which help you compose simple tunes with the inbuilt Beep.

Music Maker from Bellflower is an early attempt which didn't quite come up to scratch, editing and printer copies are virtually non existent and there are irritating pauses when a new line is needed.

Sinclair's own offering is **Musicmaster**. When it came out I said that it probably pushed this aspect of Spectrum computing to the limit and it remains a top program in this field. This has two modes of entry, note names and keyboard simulated, editing is good, replay is fast and a stave printout is provided.

And Finally . . .

from a small company (you may like to play "spot the advert" for their products) comes my favourite program so far!

MUSIC TYPEWRITER from Romantic Robot, 113 Melrose Ave. London NW2 4LX is a very professional piece of programming, easy to use, excellent screen presentation, error trapping and helpful prompts. Fast replay, versatile editing, stave and note printouts, transposing to other keys, six octaves, treble and bass clefs, 23-255MM (metronome scale), menu driven and up to 16 pieces of music held in memory at one time.

Even without an amp it soundsgood.

Do you get the impression I'm impressed?

A keyboard overlay is provided and versions of the program are available for use with a sound generator chip and even with the Midi interface if you so require.

If you are interested in music and own a computer then get it. Cost £9.95.

For those of you who enjoy DIY then watch out for our Beep amp hardware project and some musical programs soon to be published in *ZX Computing*.

Green boing!

The classic Froggy arcade game has been superbly programmed by Thomas Baker who hops around Cheshire

As in the arcade version you have to get your poor frog across the busy four lane highway and then across the fast flowing river.

The road has lorries, cars and even racing cars which you must avoid while the river has logs and turtles which you must hop on for a ride.

Once you have reached the bank at the top of the screen then you must jump into an unoccupied home, should there be a fly there at the time (a dot) then you get a bonus of 200 points.

Just in case you find this too easy there is an alligator who makes unexpected appearances in the homes, land on him and you're a nasty green blob on the landscape, as is also true should you get hit by the traffic or land in the river.

Turtles are unpredictable things and at the higher levels they may turn pink and then dive; doom to any poor frog still on them or who then tries to hop onto them.

You may also croak it if you get carried off the end of the screen or run out of time. Not an easy life being a frog!

Making it fit

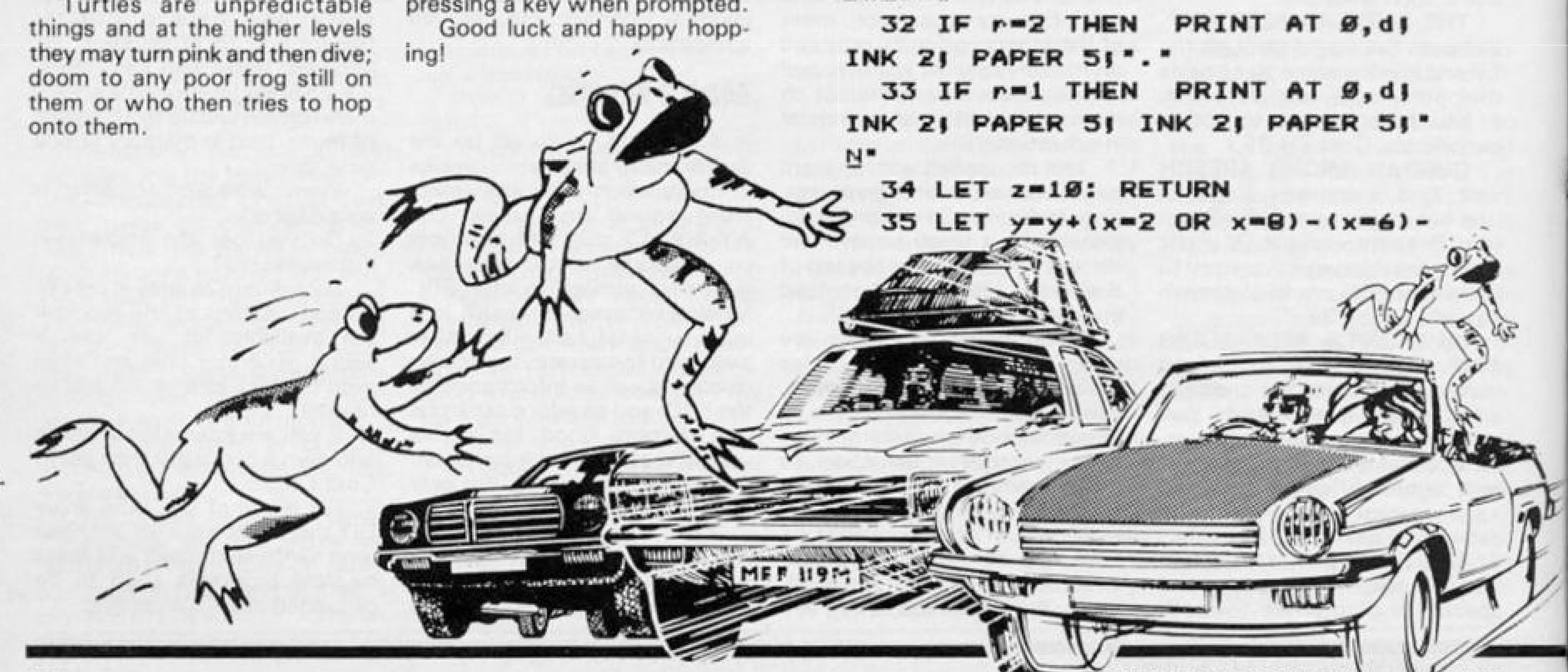
In order to get it all into 16K, it is entered in three parts, the graphics, machine code and basic.

First type in the machine code entry program, RUN it and enter one by one the numbers in the machine code list.

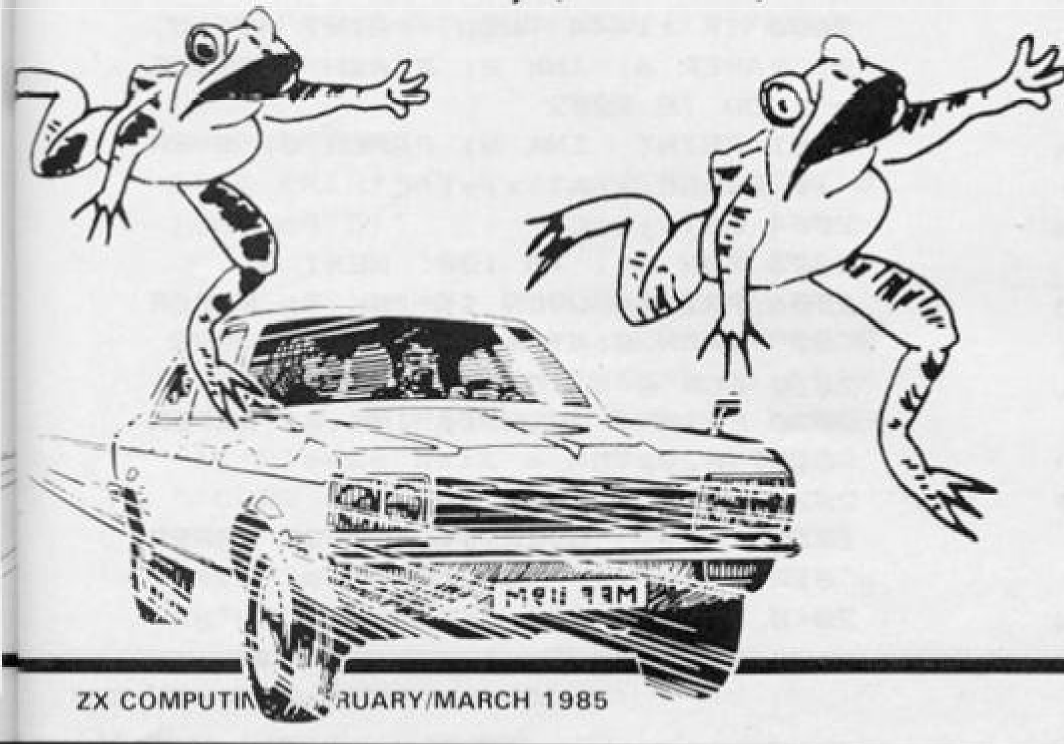
NEW the computer and type in the graphic entry program, RUN it and enter the numbers in the graphic data list. NEW the computer again and then type in the main listing. Once this has been completed then type GOTO 9999 and save the whole lot, this saves in three sections so be ready to keep pressing a key when prompted.

Good luck and happy hopping!

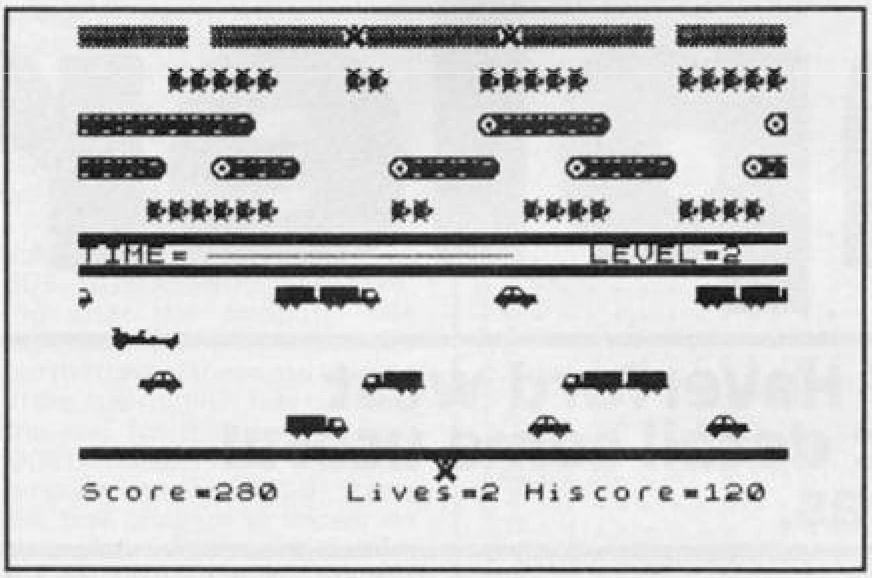
```
   1 REM ************************
         *Underlined characters*
         *are entered in        *
         *GRAPHICS mode.        *
         ************************
   2 LET hi=Ø: LET ji=Ø
   3 POKE 23658,Ø: LET a$="GGGGG
 GGGGGG GGGGGG GGGGGG GGGGG"
   4 POKE 32296,2: POKE 32308,3:
 POKE 3232Ø,1: POKE 32332,3
   5 LET h=Ø
   6 BORDER 4: INK 1: PAPER 5: C
LS
   7 LET s=Ø: LET l=3
   8 LET tc=Ø
   9 LET td=5
  1Ø PRINT AT Ø,Ø;a$
  11 GO TO 47
  12 FOR a=1 TO 1ØØØ: IF INKEY$=
"" THEN  NEXT a
  13 IF a=1ØØ1 THEN  GO TO 2
  14 LET ji=1: GO TO 2
  15 LET y=y-(x=2)-(x=8)+(x=4)+(
x=1Ø AND y<>31)
  16 RETURN
  17 LET y=y+(x=6 OR x=4)-(x=8 O
R x=2)
  2Ø RETURN
  21 IF z=Ø THEN  LET v=INT (RND
*1Ø)+1
  22 IF z<>Ø THEN  LET z=z-1: IF
 z=Ø THEN  PRINT  INK 1; PAPER 5
;AT Ø,Ø;a$: RETURN
  23 IF z<>Ø THEN  RETURN
  24 IF v>3 THEN  RETURN
  25 LET d=5+(v*7): LET r=2
  3Ø IF SCREEN$ (Ø,d)="" THEN  R
ETURN
  31 IF h>=8 THEN  LET r=INT (RN
D*2)+1
  32 IF r=2 THEN  PRINT AT Ø,d;
INK 2; PAPER 5;"."
  33 IF r=1 THEN  PRINT AT Ø,d;
INK 2; PAPER 5; INK 2; PAPER 5;"
N"
  34 LET z=1Ø: RETURN
  35 LET y=y+(x=2 OR x=8)-(x=6)-
```



```
(x=4)-(x=4)
 38 IF y<Ø THEN  LET y=Ø: GO TO
 2ØØØ
 39 IF y>31 THEN  LET y=31: GO
TO 2ØØØ
 4Ø GO TO 29Ø
 47 PRINT
 48 PRINT  INK Ø;"BBBBB    BBBB
B   BBBBB   BB    "
 49 PRINT
 5Ø PRINT  INK 2;"   QPPPPPPPQ
      QPPPPQ     "
 6Ø PRINT
 7Ø PRINT  INK 2;"QPPPQ   QPPPQ
 QPPPPQ  QPPQ   "
 8Ø PRINT
 9Ø PRINT  INK Ø;"  BBBB   BBBB
  BBBBBB    BB   "
 1ØØ PRINT  INK 6;"████████████
████████████████"
 11Ø PRINT  PAPER 6; INK 4;"
                              "
 12Ø PRINT  PAPER Ø; INK 6;"█████
███████████████████████████"
 13Ø PAPER Ø
 14Ø PRINT  INK 4;" AB         CDC
DE      CDCDE    "
 15Ø PRINT ,,
 16Ø PRINT  INK 2; BRIGHT 1;"STU
                               "
 17Ø PRINT ,,
 18Ø PRINT  INK 3;"HI          JKL
      JKLKL      "
 19Ø PRINT ,,
 2ØØ PRINT  INK 5;"AB
CDE        AB           "
 2Ø1 PRINT  INK 3;"████████████
████████████████": PRINT  PAP
ER 3,,,,
 2Ø2 PRINT AT 21,Ø; PAPER 3; INK
 Ø;"Score=       Lives=      Hi=
 "
 2Ø3 PRINT AT 21,6; PAPER 3; INK
Ø;s: PRINT AT 21,18; PAPER 3; I
NK Ø;l: PRINT AT 21,26; PAPER 3;
 INK Ø;hi
 2Ø4 IF ji=1 THEN  GO TO 215
 211 PRINT AT 1Ø,5; PAPER 6; INK
 Ø;"Press any key to start"
 212 LET t=USR 32295: LET t=USR
32344
 213 IF INKEY$="" OR INKEY$="p"
THEN  GO TO 212
 214 PRINT AT 1Ø,Ø; PAPER 6,,
 215 LET ti=184: PRINT AT 1Ø,Ø;
INK Ø; PAPER 6;"TIME=": PLOT  IN
K 3;45,91: DRAW  INK 3;139,Ø
 22Ø GO TO 8ØØ
 231 LET z=Ø
 235 PAPER 1: INK 7
 24Ø LET x=x1: LET y=y1
 25Ø PRINT  OVER 1; INK 8; PAPER
 8;AT x,y;"E"
 251 PLOT  INK 3; INVERSE 1;ti,9
1: LET ti=ti-1: IF ti=44 THEN  L
ET l=1: GO TO 2ØØØ
 252 IF h<4 THEN  GO TO 255
 253 LET tc=tc+1: IF tc=2Ø THEN
 GO SUB 5ØØØ
 254 IF tc=24 THEN  GO SUB 5Ø2Ø
 255 IF h>=2 THEN  GO SUB 21
 257 IF tc=3Ø THEN  GO SUB 51ØØ
 26Ø PRINT  FLASH Ø; OVER 1; INK
 8; PAPER 8;AT x,y;"E"
 261 IF h<8 THEN  GO TO 28Ø
 262 IF td=24 THEN  GO SUB 5Ø12
 263 IF td=3Ø THEN  GO SUB 5Ø13
 264 LET td=td+1: IF td=2Ø THEN
 GO SUB 5Ø11
 28Ø IF ATTR (x,y)=41 THEN  GO T
O 2ØØØ
 284 IF SCREEN$ (x,y)="" THEN  G
O TO 35
 285 IF x<>1Ø THEN  IF SCREEN$ (
x,y)=" " THEN  GO TO 2ØØØ
 286 IF INKEY$="" THEN  GO TO 3Ø
5
 29Ø LET y=y+(INKEY$="x" AND y<>
31)-(INKEY$="z" AND y<>Ø)
 292 IF INKEY$="l" AND x<>1Ø THE
N  GO SUB 17: LET x=x+2
 293 IF INKEY$="p" THEN  GO SUB
15: LET x=x-2: LET s=s+1Ø: PRINT
 AT 21,6; PAPER 3; INK Ø;s
 3Ø1 IF x=Ø THEN  GO TO 6ØØ
 3Ø5 LET t=USR 32344
 31Ø GO TO 25Ø
 6ØØ IF SCREEN$ (x,y)="" THEN  G
O TO 2ØØØ
 61Ø LET b$=SCREEN$ (x,y)
 62Ø IF b$="." THEN  GO TO 7ØØ
 63Ø LET a$(y+1)="E"
```



```
 635 PRINT AT Ø,Ø; PAPER 5; INK
1;a$
 636 LET h=h+1
 64Ø IF h=1 THEN  POKE 323Ø8,5:
POKE 32296,4: POKE 3232Ø,6: POKE
 32332,3
 645 IF h=2 THEN  POKE 32332,4
 646 IF h=3 THEN  POKE 323Ø8,12:
 POKE 32296,5: POKE 3232Ø,4: POK
E 32332,5
 647 IF h=5 THEN  POKE 32296,6:
POKE 3232Ø,5: POKE 32332,6
 648 IF h=6 THEN  POKE 32296,7:
POKE 3232Ø,6: POKE 32332,7
 649 IF h=7 THEN  POKE 32296,8:
POKE 3232Ø,7: POKE 32332,8
 65Ø IF h=4 THEN  POKE 323Ø8,15:
 PRINT AT 14,Ø; INK 2; PAPER Ø;"
STU         STU
"
 651 IF h=8 THEN  POKE 323Ø8,18
 652 IF h=1Ø THEN  POKE 323Ø8,2Ø
 653 IF h=12 THEN  POKE 323Ø8,23
 655 IF h=8 THEN  PRINT AT 14,Ø;
 INK 2; PAPER Ø;"STU          S
TU          STU    "
 657 IF h=14 THEN  PRINT AT 14,Ø
; INK 2; PAPER Ø;"STU      STU
 STU      STU  "
 659 IF (h/8)=INT (h/8) THEN  LE
T l=l+1
 66Ø IF (h/4)=INT (h/4) THEN  LE
T a$="GGGGG GGGGGG GGGGGG GGGGGG
 GGGGG"
 661 IF h=8 THEN  POKE 32296,9:
POKE 3232Ø,7: POKE 32332,8
 662 IF h=9 THEN  POKE 32296,1Ø:
 POKE 3232Ø,8: POKE 32332,9
 663 IF h=1Ø THEN  POKE 32296,11
: POKE 3232Ø,9: POKE 32332,1Ø
 664 IF h=11 THEN  POKE 32296,12
: POKE 3232Ø,1Ø: POKE 32332,11
 665 IF h=12 THEN  POKE 32296,13
: POKE 3232Ø,11: POKE 32332,12
 666 IF h=13 THEN  POKE 32296,14
: POKE 3232Ø,12: POKE 32332,13
 667 IF h=14 THEN  POKE 32296,15
: POKE 3232Ø,13: POKE 32332,14
 668 IF h=15 THEN  POKE 3232Ø,14
: POKE 32332,15
 669 IF h=16 THEN  POKE 3232Ø,15
 68Ø FOR v=1 TO 5: FOR m=1 TO 5:
 NEXT m: BEEP .ØØ3,Ø: BEEP .ØØ3,
1Ø: BEEP .ØØ3,2Ø: BEEP .ØØ3,3Ø:
BEEP .ØØ3,4Ø: BEEP .ØØ3,3Ø: BEEP
 .ØØ3,2Ø: BEEP .ØØ3,1Ø: BEEP .ØØ
3,Ø: NEXT v
 685 LET ti=184: PRINT AT 1Ø,Ø;
INK Ø; PAPER 6;"TIME=": PLOT  IN
K 3;45,91: DRAW  INK 3;139,Ø
 686 PRINT AT Ø,Ø; PAPER 5; INK
1;a$
 69Ø GO TO 8ØØ
 7ØØ PRINT AT Ø,y-1; FLASH 1; IN
K 2; PAPER 5;"2ØØ"
 72Ø LET a$(y+1)="E"
 725 LET s=s+2ØØ
 73Ø GO TO 636
 8ØØ LET x=2Ø: LET y=16
 81Ø PRINT AT 21,Ø; PAPER 3; INK
 Ø;"Score=        Lives=     Hi=
  "
 82Ø PRINT AT 21,6; PAPER 3; INK
 Ø;s: PRINT AT 21,18; PAPER 3; I
NK Ø;l: PRINT AT 21,26; PAPER 3;
 INK Ø;hi
 83Ø PRINT AT 1Ø,23; INK 2; PAPE
R 6;"LEVEL=";h
 835 PLOT  INK 3; INVERSE 1;ti,9
1: LET ti=ti-1: IF ti=44 THEN  L
ET l=1: GO TO 2ØØØ
 84Ø LET r$=INKEY$: LET x1=x+(2*
(r$="l" AND x<>2Ø))-(2*(r$="p"))
 85Ø IF x1=1Ø THEN  PRINT  INK 8
; PAPER 8;AT x,y;" ": LET x=x1:
GO TO 231
 855 IF r$="p" THEN  LET s=s+1Ø:
 PRINT AT 21,6; PAPER 3; INK Ø;s
 86Ø LET y1=y+(r$="x" AND y<>31)
-(r$="z" AND y<>Ø)
 87Ø IF r$="l" AND y<>2Ø OR r$="
p" THEN  BEEP .ØØ2,5: BEEP .ØØ2,
2Ø
 88Ø PRINT  INK 8; PAPER 8;AT x,
y;" "
 89Ø LET t=USR 32295
 9ØØ LET x=x1: LET y=y1
 91Ø IF SCREEN$ (x,y)="" THEN  G
O TO 2ØØØ
 92Ø PRINT  INK 8; PAPER 8;AT x,
y;"E"
 93Ø GO TO 835
2ØØØ IF ti=44 THEN  PRINT AT 1Ø,
Ø; PAPER 6; INK Ø; FLASH 1;"TIME
=": GO TO 2ØØ2
2ØØ1 PRINT  INK 8; PAPER 8; OVER
 1; FLASH 1;AT x,y;"E"
2ØØ4 LET ji=Ø
2ØØ5 FOR w=1 TO 1ØØ: NEXT w
2ØØ6 PRINT  OVER 1; INK 8; PAPER
 8; FLASH Ø;AT x,y;"E"
2Ø1Ø FOR w=x TO 2Ø
2Ø2Ø PRINT  OVER 1; INK 8; PAPER
 8;AT w,y;"E"
2Ø25 BEEP .Ø1,w
2Ø3Ø PRINT  OVER 1; INK 8; PAPER
 8;AT w,y;"E"
2Ø4Ø NEXT w
```




```
2041 LET l=l-1
2042 PRINT AT 21,18; PAPER 3; IN
K Ø;l
2043 IF l=Ø THEN  GO TO 2Ø6Ø
2045 LET ti=184: PRINT AT 1Ø,Ø;
INK Ø; PAPER 6;"TIME=": PLOT  IN
K 3;45,91: DRAW  INK 3;139,Ø
2Ø5Ø GO TO 8ØØ
2Ø6Ø PRINT AT 2Ø,11; INK 2; PAPE
R 6;"Game  Over"
2Ø61 IF s>hi THEN  LET hi=s
2Ø65 PRINT AT Ø,Ø; PAPER 4; INK
Ø; FLASH 1;"      PRESS ANY KEY T
O START       "
21ØØ GO TO 12
5ØØØ PRINT AT 8,Ø; INK 3; PAPER
5; OVER 1;"
          "
5Ø1Ø RETURN
5Ø11 PRINT AT 2,Ø; INK 3; PAPER
5; OVER 1;"
          ": RETURN
5Ø12 PRINT AT 2,Ø; INK 1; PAPER
5; OVER 1;"
          ": RETURN
5Ø13 PRINT AT 2,Ø; INK Ø; PAPER
5; OVER 1;"
          ": LET td=Ø: RETURN
5Ø25 PRINT AT 8,Ø; INK 1; PAPER
5; OVER 1;"
          "
5Ø4Ø RETURN
511Ø PRINT AT 8,Ø; OVER 1; INK Ø
; PAPER 5;"
          "
512Ø LET tc=Ø: RETURN
9998 CLEAR 32243: PAPER Ø: BORDE
R Ø: INK Ø: CLS : BRIGHT Ø: PRIN
T AT 1Ø,6; PAPER 2; INK 6; FLASH
 1;"Please wait a moment": LOAD
""CODE : LOAD ""CODE : RUN
9999 SAVE "Froggy 2" LINE 9998:
SAVE "mc"CODE 32243,149: SAVE "u
dg"CODE USR "a",168
```

Green boing machine code.
Please refer to text.

```
15,18,34,127,255,255,4Ø,16,128,6
4,32,254,254,255,4Ø,16,127,127,1
27,127,127,255,21,8,254,254,254,
254,255,255,64,128,Ø,248,196,196
,254,254,4Ø,16,153,9Ø,1Ø2,6Ø,6Ø,
126,231,66,17Ø,85,17Ø,85,17Ø,85,
17Ø,85,1,2,4,127,127,255,2Ø,8,24
Ø,72,68,254,255,255,2Ø,8,Ø,31,35
,35,127,127,2Ø,8,127,127,127,127
,255,255,2,1,254,254,254,254,254
,255,168,16,Ø,Ø,Ø,Ø,192,24Ø,255,
255,Ø,Ø,32,48,25,15,85,255,63,1Ø
3,195,153,153,195,1Ø3,63,255,14Ø
,255,255,231,252,143,255,252,254
,199,255,249,159,254,252,68,56,1
24,123,123,124,56,68,224,96,124,
255,127,255,12Ø,48,Ø,48,32,243,2
55,255,Ø,Ø,Ø,Ø,1,3,255,255,14,4
14,8,229,17,31,Ø,25,126,237,82,3
1,6,32,126,31,119,35,16,25Ø,225,
36,13,32,234,2Ø1,14,8,175,229,17
,31,Ø,237,82,126,25,23,6,32,126,
23,119,43,16,25Ø,225,36,13,32,23
3,2Ø1,6,2,197,33,128,72,2Ø5,244,
125,193,16,246,6,3,197,33,192,72
,2Ø5,244,125,193,16,246,6,1,197,
33,31,8Ø,2Ø5,13,126,193,16,246,6
,3,197,33,64,8Ø,2Ø5,244,125,193,
16,246,2Ø1,6,8,197,33,64,64,2Ø5,
244,125,193,16,246,6,16,197,33,1
59,64,2Ø5,13,126,193,16,246,6,8,
197,33,223,64,2Ø5,13,126,193,16,
246,6,8,197,33,Ø,72,2Ø5,244,125,
193,16,246,2Ø1
```

Graphic Entry Program

```
 1Ø FOR a= USR "a" TO  USR "a"+
168
 2Ø INPUT b
 3Ø POKE a,b
 4Ø NEXT a
```

Machine Code Entry Program

```
  5 CLEAR 32243
 1Ø FOR a=32244 TO 32392
 2Ø INPUT b
 3Ø POKE a,b
 4Ø NEXT a
```

Sampler

Mr W Simister of Haverford west describes in graphic detail some useful ideas.

Among the many problems facing would-be programmers — particularly those wishing to write 'games' programs — is that of furnishing the screen with suitable characters; both to reduce the emptiness of the screen, and to provide more life-like characters in the game. This is one of the simplest parts of programming the ZX Spectrum, for it requires just three things: The ability to reduce a shape to a series of numbers; a knowledge of how the FOR-NEXT loop operates; and a knowledge of how to use the PRINT-AT command.

To deal with the first requirement I have written a small subroutine which appears before the main program. enter Sampler, RUN it, and then study the checker board square in the centre of the screen. It is shown in figure 1. There are 8 squares by 8 squares. At the top are some numbers, which are read from top to bottom, and then left to right. They are 128,64,32,16,8,4,2,1. From right to left they are (1), double that (2), double that (4), double that (8), double that (16), and so on to the end. They indicate the value of any filled-in square in the column below them.

128	64	32	16	8	4	2	1	
1	1	1						224
			1					16
		1						32
	1	1	1	1	1			124
	1	1	1	1	1	1		126
		1	1	1	1	1	1	63
				1	1	1	1	15
					1			4

Fig.1. Sampler character design grid.

The numbers to the right of the squares are the totals of the filled-in squares in that row. The first (top) row is 224, which is the total of 128+64+32. The second row is 16, and so on down the right hand side of the square.

Those figures on the right hand side are the ones used (from top to bottom) in the DATA line. They are entered as you will see in the program at line 300. To list the program again use BREAK, and then LIST ENTER. There are programmers who use the 'BIN 11100000' format for their DATA lines, but they are harder to enter. Study the difference between: 160 DATA 224,16,32,124,126,63,15,4 — and — DATA BIN 11100000, BIN 00010000, BIN 00100000, BIN 01111100, BIN 01111110, BIN 00111111, BIN 00001111, BIN 00000100. Either way will work, but I prefer to use decimal numbers. However, using DATA BIN the computer works slightly faster.

LOOPS

FOR-NEXT loops are equally simple to understand. To make a graphic character we use 4 commands: FOR x=0 TO 7:(which means x stands for the 8 rows of 8 squares; READ y: (this means the place where the computer will store the character); POKE USR "A"+x,-y: (means that A will be used to stand for the graphic character — but only when it is entered in the graphic mode); NEXT x tells the computer to go back to the previous 'FOR x' position and repeat the loop a further 7 times, collecting the DATA for each row in succession until all 8 rows are filled.

REM A (or B,C, or D) is a reminder to yourself which letter you used for this character, because after the program has been run once, the letter will have taken over the character of the DATA you typed in. If you are using a lot of characters you may forget them. Such REM reminders must only be at the end of a line (unless on their own) for the computer will ignore anything in the line after REM.

A different set of FOR-NEXT loops is shown in lines 5070 TO 5085. There they are used to draw the lines forming the 8×8 square. Those two loops should be compared for their differences, for one draws horizontal lines from left to right, and the other draws the vertical lines from bottom to top. The horizontal lines are governed by: 5070 LET x=64: For y=48 TO 112 STEP 8 and by: 5075 PLOT x,y: Draw 64,0: NEXT y

First of all it must be understood that PLOT-DRAW commands make the computer look for a 'pixel' number. In each one of the squares, in the 8×8 square we are working on, are 8 rows of 8 pixels in each row. PRINT-AT commands (which will be explained later) deal with the 704 position on the complete screen. PLOT-DRAW commands deal with the 64 pixels (points that can be drawn) in each of the 704 positions (44625 points in all the complete screen). The bottom (or x) columns of pixels run from 0 to 255, left to right. The right hand side (or y) rows of pixels run from 0 to 175, bottom to top. Therefore PLOT x,y: means position pixel 64, pixels 48 TO 112 8 steps apart (because in the previous line x and y had been given those values). The STEP 8 means that when a line has been drawn the next line is to be 8 pixels from that, and so on until the entire 48 to 112 has been used up. That repetition has been put into effect by NEXT y, which commands the computer to go back to the first line until all the plots between 48 to 112 have been filled.

The position of those numbers each side of the comma is very important. They reflect the x,y position. 64,0 means that a horizontal line is drawn from LEFT to right; 0,64 means that a vertical line is drawn from bottom to top, −64,0 would produce a horizontal line from RIGHT to left; and 0,−64 would produce a vertical line from top to bottom. Draw 64,64 would instruct the computer to draw a line diagonally upwards to the right.

Other commands

That is enough about FOR-NEXT loops for the moment. Now press P for the second SUBROUTINE: the PRINT-AT commands. Study the screen well (shown in figure 2). The position where any letter/character/number can be placed is indicated by a single dot. Look at row 12 (line 5570). Each dot in that statement could be replaced by a space, for spaces enclosed by double quotes are always recognised by the computer. I have used a dot between ' ' and 'first', but spaces between 'first' & 'OR' & 'x'. Alternatively, if there were a lot of

spaces to insert, the line could read: 5570 PRINT AT 12,0; ''12''; AT 12,5; '' (s) first(s)or(s)x''. In this context (s) stands for a space. Notice the semi-colon before AT in the second part of the line.

Programmers interested in sub-routines should study line 505: GOSUB 5000. This, coming after the computer has digested the long list of graphics (so that one of these can be used in the sub-routine) tells it to skip the rest for the moment, and GOTO 5000. Now scroll the program on to line 5000. This is the first program to appear on the screen. At line 5095 it instructs the computer to print ''Press P for next program'': PAUSE 5000.

This pause is about 1 ½ minutes long (50 is about 1 second) and ensures that the user will have some time to study the screen. If you need longer, then alter that pause to 50000, and you'll then have 16 ½ minutes.

Line 5100 brings in the INKEY$ command. IF INKEY$ = ''P'' (this means that if you press P) THEN CLS: (which means clear the screen). Then comes GOTO 5500 (next program). Notice the use of 'OR': IF INKEY$ = ''P'' OR INKEY$ = ''p''. This ensures that if you press either P or p it will proceed to 5500. In this second sub-routine the end line, after clearing the screen, says simply:RETURN. At this command (which is part of a GOSUB-RETURN routine) the computer goes back to the line after the GOSUB command. So it goes back to 600 (the line after 505 GOSUB 5000) and this is the main program. The scene is set with bushes and trees in green, and the animals are put in afterwards (so that they can be a different colour). In the PRINT-AT commands for the bushes a space (or spaces) is used between characters. In the PRINT-AT commands for the animals (lines 810 to 840) they are placed at specific points to avoid any spaces blotting out previously placed green bushes. The use of a plotting board is almost essential when placing characters made of more than 1 letter (like the larger elephant). I got mine from J.S.E. of Bradford on Avon. These four letters, MNLO, had to be placed on 2 lines: MN above LO. You will see that this is done in lines 825, 830 and 835, 840.

```
0123456789 & on to 31..........
1..............................
2....second or Y...............
3..............................
4..............................
5....SCREEN X,Y POSITIONS......
6..............................
7..............................
8..............................
9..............................
10.............................
11.............................
12..:.first or X...............
13.............................
14.............................
15.....you can position........
16.....anything anywhere.......
17.............................
18.............................
19.............................
20.............................
21..press M for main program...
```

Fig.2. Example Screen of Sampler PRINT AT option.

Motion

The first movement in the program is now started in line 1010. It provides a good example of nested FOR-NEXT loops: a is outside two other loops; b to make elephant ''C'' travel 2 journeys, and c to make him travel from left to right. In line 1015 the elephant (now A$) is printed at row 12, c times (0 TO 31), and behind him spaces are printed to prevent a row of elephants being left on the screen. Now, in between these two print statements, is PAUSE 3: This should be studied. Without that pause the elephant would flicker across the screen three times as fast. Note the slight difference in movement of the second elephant (from right to left). That has PAUSE 2. Try altering either of these pauses to 1 or 5, and note the difference. This is a good way to control speed.

To continue with the FOR-NEXT loops. After the PRINT-AT statements is NEXT c: PAUSE 20: NEXT b. The pause of 20 (less than ½ a second) allows a tiny interval between its disappearance on the right, and its reappearance on the left. You could use PAUSE 30 instead.

NEXT b directs the computer back to line 1010, and it finds that the elephant must repeat its journey. It does that, and then, being fulfilled, it goes on to the next line. At 1020 a second set of FOR-NEXt loops (e and f), make the elephant travelling from right to left make 2 journeys also. Then, with loop a still uncompleted, it returns to 1010, and does it all again, so that loop a also is completed. Then it passes on to the next section.

It is well worthwhile to enter RUN 1000 ENTER to watch this on its own. Any section of this entire program can be selected in a similar way to study it. My usual practice when working on a section is to LIST 1000 (or whichever line I want). It is simple then to bring down each line in turn, convert its number to one well beyond the program — say 9000 — and work on it there without upsetting the main program. When finished, the lines can be obliterated (9000 ENTER and so on) or they can be used to replace the lines they came from by renumbering them back to their original numbers.

In order to study the way in which FOR-NEXT loops can be nested look at figure 3 which is an example of the lines 1010 to 1025, rewritten to separate out all the sections of a line that are separated by a colon. You will see that brackets have been used to join together all the FOR-NEXT loops according to their letter. Letter 'a' encloses the lot,

```
9000  FOR a = 0 TO 1
9001  FOR b = 0 TO 1
9002  FOR c = 0 TO 31
9003  LET A$ = "C"
9004  PRINT AT 12,c; A$
9005  PAUSE 3
9006  PRINT AT 12,c; " "
9007  NEXT c
9008  PAUSE 20
9009  NEXT b
9010  FOR e = 0 TO 1
9011  LET B$ = "B"
9012  FOR f = 31 TO 0 STEP -1
9013  PRINT AT 14,f; B$
9014  PAUSE 2
9015  PRINT AT 14,f; " "
9016  NEXT f
9017  PAUSE 20
9018  NEXT e
9019  PAUSE 20
9020  NEXT a
```

Fig.3. Nested FOR-NEXT loop structure.

while inside 'a' are two nested loops: 'c' inside 'b'., and 'f' inside 'e'. Study this, and watch it working. You should never again be puzzled by FOR-NEXT loops.

The next movement, lines 1500 to 1550, is a flock of birds flying for right to left. Their movement is controlled by the Spectrum 'String Slicing' technique (an explanation of which is on page 51 of the manual). Of considerable importance is the grouping of the birds to make a flock.

The DATA for bird "H" in line 260 has three 0s at its end, thus lifting the bird's image to the top of the 8 × 8 square of pixels. If another letter is chosen to carry another graphic (in this case "J" in line 280) and those three 0s are transferred to the beginning of the DATA line for that bird, then it will appear at the bottom of the pixel square. Quite a difference in level. A judicious sprinkling of H&J (in graphic mode, of course) along line 1505 gives two levels of bird: 1505 LET H$ = "HJ J H JH J H J JH – and so on". In line 1520, the PRINT-AT line, we can now print three rows of H$ or J$ or both (see line 1510 LET J$ = H$) and produce a flock of birds. The actual movement is controlled by lines 1525 and 1530. Lines 1540 to 1550, printing "spaces", are there to clear the screen before the next display., Using CLS here would clear away the bushes and trees.

More wildlife

Lines 2000 to 2065 are an amusing interjection, again using the string slicing method. A deer runs through the wood, left to right, startling a large running bird that doubled back towards the left. In this program it is repeated with a FOR-NEXT loop (0 TO 5), so that the user can study it, but in actual use it could be just a single instance (by leaving out the FOR-NEXT loop).

In Lines 2500 to 2535 the program reverts to the original slicing technique, but with a single bird this time. It is a fairly smooth sort of movement, and a careful comparison of line 2525 with lines 1525 and 1530 may well give the user other ideas for its use.

The final movement in the program is achieved in lines 3000 to 3060. It is a frenzied running backward and forward by two deer. Without the FOR-NEXT loop (0 TO 4) the two deer would make only one visit (there and back). With an increased number in line 3020 (0 TO 500?) they would continue that a number of times. You would be tired long before they would.

Finally...

At the end of the program the graphics used are displayed on the screen to signal the end. But it need not be the end for you. Remember that all the sections will run separately, so what is to stop you extracting a section, making different graphics for it (and the possibilities in that line are enormous) and so producing a simple game. This program is called a Sampler, in imitation of the Victorian ladies who produced canvases on which were many kinds of stitch — for their future reference. I hope you will find this sampler useful.

```
   1 REM ***********************
        *Underlined characters*
        *are entered in       *
        *GRAPHICS mode.       *
        ***********************
  10 REM Sampler for programmers
********************************
  20 REM W Simister ************
********************************
  30 REM This POKE makes the key
s BEEP ************************
  40 POKE 23609,255: BORDER 5: I
NK Ø
 200 REM Making single graphics
********************************
 205 FOR x=Ø TO 7: READ y: POKE
USR "A"+x,y: NEXT x: REM A Cocke
rel
 210 DATA 48,225,49,62,62,28,20,
34
 215 FOR x=Ø TO 7: READ y: POKE
USR "B"+x,y: NEXT x: REM B Eleph
ant
 220 DATA Ø,48,56,126,95,159,18,
18
 225 FOR x=Ø TO 7: READ y: POKE
USR "C"+x,y: NEXT x: REM C Eleph
ant
 230 DATA Ø,12,28,126,25Ø,249,72
,72
 235 FOR x=Ø TO 7: READ y: POKE
USR "D"+x,y: NEXT x: REM D Tree
 240 DATA 16,56,56,124,124,254,1
6,16
 245 FOR x=Ø TO 7: READ y: POKE
USR "E"+x,y: NEXT x: REM E Deer
 250 DATA 16,32,64,224,127,63,5Ø
,82
 255 FOR x=Ø TO 7: READ y: POKE
USR "H"+x,y: NEXT x: REM H
 260 DATA 4,72,223,62,8,Ø,Ø,Ø
 265 FOR x=Ø TO 7: READ y: POKE
USR "I"+x,y: NEXT x: REM I
 270 DATA 8,4,2,7,254,252,76,74
 275 FOR x=Ø TO 7: READ y: POKE
USR "J"+x,y: NEXT x: REM J
 280 DATA Ø,Ø,Ø,4,72,223,62,8
 285 FOR x=Ø TO 7: READ y: POKE
USR "K"+x,y: NEXT x: REM K
 290 DATA Ø,Ø,2,7,126,252,76,146
 295 FOR x=Ø TO 7: READ y: POKE
USR "P"+x,y: NEXT x: REM P
 300 DATA 224,16,32,124,126,63,1
5,4
 305 FOR x=Ø TO 7: READ y: POKE
USR "S"+x,y: NEXT x: REM S
 310 DATA 24,126,255,255,255,126
,24,24
 400 REM Making larger graphics
********************************
*
 405 FOR x=Ø TO 7: READ y: POKE
USR "F"+x,y: NEXT x: REM F
 410 DATA 16,96,224,16Ø,32,32,48
,56'
 415 FOR x=Ø TO 7: READ y: POKE
USR "G"+x,y: NEXT x: REM G
 420 DATA 61,63,63,55,51,51,99,9
9
 425 FOR x=Ø TO 7: READ y: POKE
USR "L"+x,y: NEXT x: REM L
 430 DATA Ø,Ø,Ø,Ø,Ø,31,63,127
 435 FOR x=Ø TO 7: READ y: POKE
USR "M"+x,y: NEXT x: REM M
 440 DATA Ø,Ø,112,24Ø,248,252,25
2,252
 445 FOR x=Ø TO 7: READ y: POKE
USR "N"+x,y: NEXT x: REM N
 450 DATA 255,255,255,255,113,96
```

```
,96,96
 455 FOR x=Ø TO 7: READ y: POKE
USR "O"+x,y: NEXT x: REM O
 460 DATA 222,206,199,195,195,19
8,192,192
 465 FOR x=Ø TO 7: READ y: POKE
USR "Q"+x,y: NEXT x: REM Q
 470 DATA 120,255,31,15,24,40,40
,24
 475 FOR x=Ø TO 7: READ y: POKE
USR "R"+x,y: NEXT x: REM R
 480 DATA Ø,224,240,248,100,82,8
1,208
 485 FOR x=Ø TO 7: READ y: POKE
USR "T"+x,y: NEXT x: REM T
 490 DATA 24,60,60,126,126,126,2
55,255
 495 FOR x=Ø TO 7: READ y: POKE
USR "U"+x,y: NEXT x: REM U
 500 DATA 255,255,255,255,126,60
,24,24
 505 GO SUB 5000
 600 REM Now set the scene *****
********************************
 605 REM Note where colour (INK
4) is introduced ***************
********************************
 610 PRINT AT 10,Ø; INK 4;"S  S
 D   S  S D   S   S  D"
 615 PRINT AT 11,1; INK 4;"D   S
 D   S D   S D  S   D  S S"
 620 PRINT AT 13,Ø; INK 4;"S    S
 D   D    D S   S D  S  S"
 625 PRINT AT 15,2; INK 4;"D S
 D   D S  D  S  D     T D"
 630 PRINT AT 16,1; INK 4;"D
T  S      D      D  U"
 635 PRINT AT 17,4; INK 4;"D UD
  D  S    D   S  D S"
 640 PRINT AT 18,2; INK 4;"S
  D   T     S"
 645 PRINT AT 19,Ø; INK 4;"D
D    S U  D   D T D   D"
 650 PRINT AT 20,2; INK 4;"T
S  D  D      D    U  S"
 655 PRINT AT 21,2; INK 4;"U S
 D    S  D S  D S D   DS"
 800 REM Now place the still ani
mals ***************************
 805 REM Some with one graphic l
etter and others with more *****
********************************
 810 PRINT AT 15,3;"K";AT 15,5;"
K"
 815 PRINT AT 16,4;"K";AT 16,9;"
T";AT 16,27;"E"
 820 PRINT AT 17,2;"T";AT 17,23;
"E";AT 17,27;"G"
 825 PRINT AT 18,6;"LM";AT 18,24
;"E";AT 18,28;"QR"
 830 PRINT AT 19,6;"NO";AT 19,15
;"T";AT 19,22;"E"
 835 PRINT AT 20,5;"LM"
 840 PRINT AT 21,5;"NO";AT 21,18
;"A";AT 21,21;"P"
1000 REM FOR-NEXT loops *******
1005 REM Note the pauses *******
********************************
1010 FOR a=Ø TO 1: FOR b=Ø TO 1:
 FOR c=Ø TO 31
1015 LET A$="C": PRINT AT 12,c;A
$: PAUSE 3: PRINT AT 12,c;" ": N
EXT c: PAUSE 20: NEXT b
1020 FOR e=Ø TO 1: LET B$="B": F
OR f=31 TO Ø STEP -1: PRINT AT 1
4,f;B$: PAUSE 2: PRINT AT 14,f;"
 ": NEXT f
1025 PAUSE 20: NEXT e: PAUSE 20:
 NEXT a
1500 REM Using Spectrum slicing
method for a string ************
********************************
1505 LET H$="HJ  H  JHJ  H   HH
   JHJ  JH    "
1510 LET J$=H$
1515 FOR x=Ø TO 50
1520 PRINT AT 3,Ø; INK 1;H$;AT 4
,Ø;J$;AT 2,Ø;J$(3 TO  )+J$(1 TO
2)
1525 LET H$=H$(2 TO  )+H$(1)
1530 LET J$=J$(3 TO  )+H$(1 TO 2
)
1535 NEXT x
1540 PRINT AT 2,Ø;"
                               "
1545 PRINT AT 3,Ø;"
                               "
1550 PRINT AT 4,Ø;"
                               "
2000 REM Slicing in 2 directions
********************************
2005 LET B$="
          II "
2010 FOR x=Ø TO 5
2015 LET l=LEN B$
2020 FOR i=2 TO (l-1)
2025 PRINT AT 12,Ø;B$((l-i) TO l
)
2030 PRINT AT 12,26;"      "
2035 NEXT i
2040 LET C$="
          A "
2045 LET l=LEN C$
2050 FOR j=(l-1) TO 2 STEP -1
2055 PRINT AT 14,Ø;C$((l-j) TO l
)
2060 PRINT AT 14,Ø;"  "
2065 NEXT j: NEXT x
```

```
2500 REM Slicing ***************
********************************
2505 FOR x=0 TO 178
2510 LET D$="
       H "
2515 FOR x=0 TO 178
2520 PRINT AT 9,0;D$
2525 LET D$=D$(2 TO  )+D$(1)
2530 NEXT x
2535 PRINT AT 9,0;"
                 "
3000 REM Slicing ***************
3005 REM Hurried to & fro moveme
nt *****************************
3010 LET a$="
     II "
3015 LET b$="
      EE "
3020 FOR x=0 TO 4
3025 LET l=LEN a$
3030 FOR i=0 TO (l-1)
3035 PRINT AT 14,0;a$((l-i) TO l
)
3040 NEXT i
3045 FOR j=(l-1) TO 0 STEP -1
3050 PRINT AT 14,0;b$((l-j) TO l
)
3055 NEXT j
3060 NEXT x: CLS : PAUSE 20
4000 REM Finishing statement ***
********************************
4010 PRINT AT 10,1;"These are th
e graphics in this  program"
4020 PRINT AT 20,1;"A B C D E H
I J K P QR S"
4030 PRINT AT 17,5;"F LM T"
4040 PRINT AT 18,5;"G NO U"
4045 STOP
5000 REM Making USR graphics****
********************************
5015 PRINT AT  5,8;"1
 MAKING"
5020 PRINT AT  6,8;"2631
 USR"
5025 PRINT AT  7,8;"84268421
 GRAPHICS"
5030 PRINT AT  8,8;"███      224"
5035 PRINT AT  9,8;"   █      16"
5040 PRINT AT 10,8;"  █       32"
5045 PRINT AT 11,8;" █████   124"
5050 PRINT AT 12,8;" ██████  126"
5055 PRINT AT 13,8;"  ██████63"
5060 PRINT AT 14,8;"    ████15"
5065 PRINT AT 15,8;"     █   4"
5070 LET x=64: FOR y=48 TO 112 S
TEP 8
5075 PLOT x,y: DRAW 64,0: NEXT y
5080 FOR x=64 TO 128 STEP 8: LET
 y=48
5085 PLOT x,y: DRAW 0,64: NEXT x
5090 PRINT AT 17,0;"PP P  P   PP
P PP   P   P   PPPP "
5095 PRINT AT 20,0;"Press P for
next program": PAUSE 5000
5100 IF INKEY$="P"  OR INKEY$="p
" THEN  CLS : GO TO 5500
5500 REM Board plotting position
s*******************************

5510 PRINT AT 0,0;"0123456789 &
on to 31.........."
5515 PRINT AT 1,0;"1......^....
..................."
5520 PRINT AT 2,0;"2.....second
or Y..............."
5525 PRINT AT 3,0;"3...........
..................."
5530 PRINT AT 4,0;"4...........
..................."
5535 PRINT AT 5,0;"5.....SCREEN
X,Y POSITIONS......"
5540 PRINT AT 6,0;"6...........
..................."
5545 PRINT AT 7,0;"7...........
..................."
5550 PRINT AT 8,0;"8...........
..................."
5555 PRINT AT 9,0;"9...........
..................."
5560 PRINT AT 10,0;"10.........
...................."
5565 PRINT AT 11,0;"11.........
..................."
5570 PRINT AT 12,0;"12..<.first
or X..............."
5575 PRINT AT 13,0;"13.........
..................."
5580 PRINT AT 14,0;"14.........
..................."
5585 PRINT AT 15,0;"15.....you c
an position........"
5590 PRINT AT 16,0;"16.....anyth
ing anywhere........"
5595 PRINT AT 17,0;"17.........
..................."
5600 PRINT AT 18,0;"18.........
..................."
5605 PRINT AT 19,0;"19.........
..................."
5610 PRINT AT 20,0;"20.........
...................."
5615 PRINT AT 21,0;"21..press M
for main program....": PAUSE 500
0
5620 IF INKEY$="M" OR INKEY$="m"
 THEN  CLS : RETURN

5630 GO TO 5620
```

University Software

UNISTAT

STATISTICAL PACKAGE

MULTIVARIATE REGRESSION: Main output consists of estimated coefficients, t-stats, standard errors, R^2, corrected R^2, standard error of regression, F stat. and Durbin-Watson stat. Further output options: multiple correlation matrix, var-covar matrix, ANOVA of regression, residuals, plot of residuals, interpolation. Data options: Log/ln option for each variable, no-constant regression, choice of dependent variable, selection of independent variables, auto-omission of linearly dependent variables.

ANALYSIS OF VARIANCE, SCATTER DIAGRAMS AND TIME SERIES PLOTS: One-way and two-way (without interaction) ANOVA tables, scatter diagrams of paired data and time series plots with auto-scaling.

STATISTICAL TESTS, CORRELATION COEFFICIENTS AND PROBABILITY DISTRIBUTIONS: Basic stats on each column of data (size, sum, mean, variance, std. dev.). Chi-square (contingency table), t (one sample, two sample, paired), F, Mann-Whitney U, and Wilcoxon signed rank tests. Pearson's, Spearman's rank and Kendall's rank corr. coefs. Chi-square, t, F, binomial, Poisson and normal (std., non-std.) distributions. Results of tests and corr. coefs are displayed with significance levels.

DESCRIPTIVE STATISTICS, FREQUENCY DISTRIBUTIONS AND HISTOGRAMS: Analysis of raw data or data with frequency counts. Raw data sorted and grouped. Choice of lower bounds and class intervals. Absolute, cumulative and relative frequencies. Histograms with up to 200 classes. Output displays sum, mean, mean deviation, median, variance, std. dev., 3rd and 4th moments, skewness, kurtosis, range, etc.

All programs are datafile compatible with Matrix Operations program in UNIMAX package. Data matrix capacity examples (columns by rows): **CBM-64;** 2x1250, 5x700, 10x380, **BBC-B;** 2x750, 5x400, 10x200, **48K Spectrum;** 2x1800, 5x900, 10x500. **CBM-64** (disk/cass): **£85, BBC-B** (disk/cass): **£85, 48K Spectrum** (mdvcart/cass): **£60**

UNIMAX

LINEAR PROGRAMMING PACKAGE

LINEAR PROGRAMMING: A powerful and instructive optimisation program capable of handling all sorts of linear programming problems (min/max, any combination of <=, =, >= constraints and $x_i>=0$, $x_i<=0$, $-\infty<x_i<\infty$ sign constraints). Primal, canonical, dual and their solutions are displayed in standard mathematical form. Unbounded problem and no feasible solution prompts. Edit option for all inputs. Capacity examples (variables by constraints): **CBM-64;** 10x35, 25x30, 40x25, **BBC-B;** 10x25, 20x20, **48K Spectrum;** 10x45, 25x35, 50x25.

MATRIX OPERATIONS: Inversion, transposition, determinant, pre- and post-multiplication, scalar multiplication, addition and subtraction of matrices and vectors. Any output can in turn be used as the input of the next operation without re-typing. Matrices can be saved or loaded at any stage. Datafile compatible with **UNISTAT** package. Capacities: **CBM-64;** 35x35, **BBC-B;** 25x25, **48K Spectrum;** 45x45.

CBM-64 (disk/cass): **£60, BBC-B** (disk/cass): **£60, 48K Spectrum** (mdvcart/cass): **£45.**

ALSO AVAILABLE FOR 48K SPECTRUM
INTRODUCTION TO ECONOMICS SET: £25
ECONOMIC GEOGRAPHY: £9.95 • BIBLIOFILE: £9.95
POLYNOMIALS: £6.95 • INTEGRATION: £6.95

To: **UNISOFT LIMITED**
(Dept. A) 29 ST PETER'S STREET, LONDON N1 8JP.
TEL: 01-359 0978

Prices include VAT and delivery within the UK. For orders from Europe add £1.50, outside Europe £3.00 per item for airmail post. Dealers enquiries welcome.

COMMODORE 64, BBC-B, 48K SPECTRUM

✻ Comprehensive user manual ✻ Full data handling facilities (display, print, change, save, load)
✻ All results can be printed out ✻

It's easy to complain about advertisements. But which ones?

Every week millions of advertisements appear in print, on posters or in the cinema.

Most of them comply with the rules contained in the British Code of Advertising Practice.

But some of them break the rules and warrant your complaints.

If you're not sure about which ones they are, however, drop us a line and we'll send you an abridged copy of the Advertising Code.

Then, if an advertisement bothers you, you'll be justified in bothering us.

The Advertising Standards Authority.
If an advertisement is wrong, we're here to put it right.

ASA Ltd, Dept 2 Brook House, Torrington Place, London WC1E 7HN

This space is donated in the interests of high standards of advertising.

Conversion tips

A guide to ZX81/Spectrum program conversions from David Nowotnik.

The versions of BASIC offered by the two ZX computers are so similar that many programs for one can be used by the other. The ZX81 has only two commands which are not present on the Spectrum, SCROLL and UNPLOT, and these should cause you few problems when converting ZX81 programs to the Spectrum (see Table 1).

There are quite a lot of commands and functions on the Spectrum which are not available on the ZX81. A list of these appears in Table 4. The stars indicate those commands and functions for which there is no simple translation to ZX81 BASIC. Those for colour and sound can be ommitted; but you will have to find some alternative for the high resolution and file I/O commands.

The command PLOT appears on both computers, but the effect is quite different, so beware! Another tip: PEEK and POKE should be used with caution. In conversion, addresses will almost certainly have to be changed. Some of those changes appear in the tables. A command such as POKE USR "a".... on the Spectrum indicates User Defined Graphics; ZX81 users don't have this facility, so you'll have to omit this and use a standard character instead.

ZX81	Spectrum	Comments
SCROLL	RANDOMISE USR 3582 or LET t=USR 3582	If the program uses random numbers, they could become rather predictable with the first option. If so, use the second, using a variable (in this case t) which is otherwise not used.
PLOT Y,X	PRINT AT 21−Y/2,X/2;	Print the appropriate quarter square graphics character.
UNPLOT Y,X	PRINT AT 21−Y/2,X/2;	Print a space, or the appropriate quarter square graphics character.

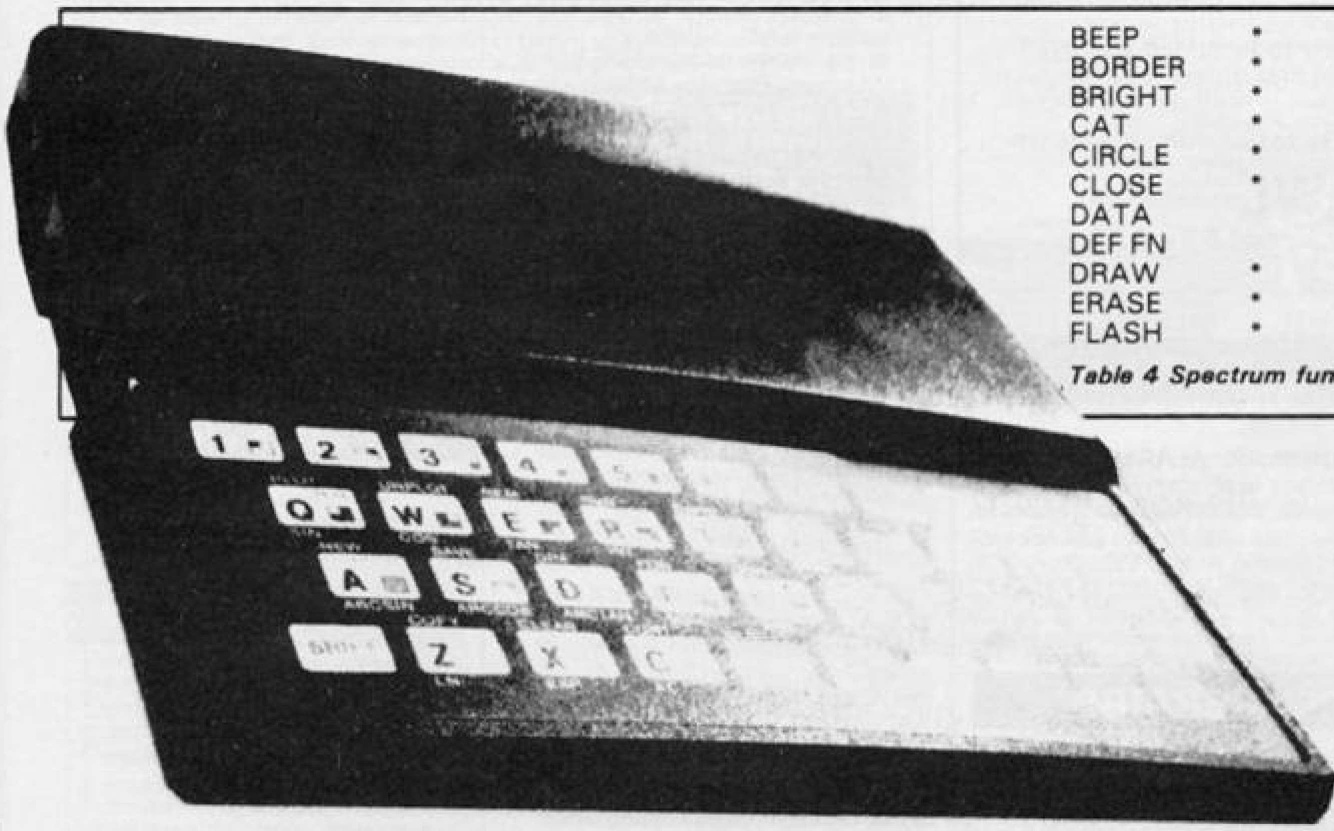
Table 1 ZX81 to Spectrum conversions.

Spectrum	ZX81	Comments
BIN eg LET y=BIN 10010101	LET y=(decimal no.) Conversion to decimal: 10010101=149 128 64 32 16 8 4 2 1 Add these numbers together when a 1 appears at the appropriate position in binary.	BIN allows the representation of a number in binary. On the ZX81 use the decimal equivalent, but beware; BIN is often used with User Defined Graphics, which are not available on the ZX81.
READ/DATA eg READ x,y DATA 50,60	LET LET X=50 LET Y=60	READ and DATA are used to store a lot of information in a program. Use LET instead.
DEF FN and FN eg DEF a(x)=SQR x LET t=FN a(i)	LET X$="SQR X" LET X=I LET T= VAL X$	The defined function can appear in a string. Use the keyword for built-in functions (eg SQR). The equivalent of FN may need 2 lines, as shown.
PLOT	no equivalent	
SCREEN$ eg LET a=SCREEN$ x,y	LET A=PEEK(PEEK 16396 +256*PEEK 16397+1+Y+33*X)	Used in interactive games to detect characters in the display file. Note — this formula only works when a RAM pack is fitted.

Table 2 Spectrum to ZX81 conversions.

ZX81	Spectrum	Comments
1 FRAMES POKE 16436,255 POKE 16437,255 LET T=(65535 - PEEK 16436 - 256*PEEK 16437) /50	POKE 23672,0:POKE 23673,0 LET t=(PEEK 23672+256* PEEK 23673)/50 For times greater than 10 minutes, you can use byte 23674 as well.	Both computers have a counter which accurately varies by 50 every second. In the example, use the first line to start the 'clock'. The variable T will have the time in seconds after the start. The counter can only be used for 10 minutes.
2 Line number zero POKE 16510,0	POKE 23756,0 (As the start of BASIC can move, eg with microdrives) use with caution.	Converts the first line of a program to line number zero, which cannot be edited, and so is protected.
3 RAMTOP POKE 16388,X - 256*INT (X/256) POKE 16389, INT (X/256)	CLEAR x	Creates a safe area at the top of RAM starting at address x, for storing data, machine code etc.

Table 3 General interconversion hints.



BEEP	*	FORMAT	*	ATTR	*
BORDER	*	INK	*	BIN	
BRIGHT	*	INVERSE		FN	
CAT	*	MERGE	*	IN	*
CIRCLE	*	MOVE	*	OVER	*
CLOSE	*	OPEN	*	POINT	*
DATA		OUT	*	SCREEN$	
DEF FN		PAPER	*	VAL$	*
DRAW	*	READ			
ERASE	*	RESTORE	*		
FLASH	*	VERIFY	*		

Table 4 Spectrum functions not available on the ZX81.

System Variables Conversion Table.

Variable	ZX81/ T/S1000	Spectrum/ TS2068
BREG	16414	23655
CDFLAG	16443	No Equivalent
CH ADD	16406	23645
COORDS	16438	23677
COORDS (Byte 2)	16439	23678
DEST	16402	23629
DF CC	16398	23684
D FILE	16396	No Equivalent
DF SZ	16418	23659
E LINE	16404	23641
ERR NR	16384	23610
E PPC	16294	23625
ERR SP	16386	23613
FLAGS	16385	23611
FLAGX	16429	23665
FRAMES	16436	23672
LAST K	16421	23560
MARGIN	16424	No Equivalent
MEM	16415	23656
MEMBOTT	16477	23698
MODE	16390	23617
NXTLIN	16425	23637
OLDPCC	16427	23662
PPC	16391	23621
PRBUFF	16444	23296
PR CC	16440	23680
RAMTOP	16388	23730
SEED	16434	23670
S PSN	16441	23688
S POSN (Byte 2)	16442	23689
STKBOT	16410	23651
STKEND	16412	23653
S TOP	16419	23660
STRLEN	16430	23666
T-ADDR	16432	23668
VARS	16400	23627
VERSN	16393	No Equivalent
X PTR	16408	23647

ZX COMPUTING

Lineage: 40p per word.
Semi display: £9.00 per single column centimetre. Ring for information on series bookings/discounts.

All advertisements in this section must be prepaid.
Advertisements are accepted subject to the terms and conditions printed on the advertisement rate card (available on request).

01 - 437 0699

Send your requirements to:
JASON INSKIP
ASP Ltd., 1 Golden Square,
London W1

SOFTWARE GAMES

LIBRARIES

ALARMS

REPAIRS

WANTED

COURSES

SOFTWARE APPLICATIONS

SOFTWARE APPLICATIONS

SOFTWARE APPLICATIONS

MICRODRIVE COMPATIBLE
. . . presenting the FLEXI FILER

"User definable record format ensures no wasted space * 30K capacity per file. Up to 1500 records on Spectrum 48K * Up to 15 fields per record — max 56 chrs per field * Update, sort, maintain, review, print file * Search routine works for any field * Totalise numeric fields * 100% microdrive compatible. Save to cartridge or cassette * Fully menu-driven for easy operation; flex-filer; ideal business/domestic filing system. 102 uses. Supplied double-recorded + full instructions. **£6.95** inc.

S. D. Micro-Systems (ZX), PO Box 24, Hitchin, Herts.

Return this ad for 10% discount. SAE for full list of utility programs.

SOFTWARE EDUCATIONAL

KILSOFT

Educational Revision/Tutorial Programs 48K Spectrum

GCE/CSE Biology, Physics, Chemistry, Geog, Maths and French. **For Home & School Use.** Only **£4.95** each. Be well prepared. We also have programs from **£2.50**. For fullest details write today. (Astronomy — Now only **£6.95**)

32 Brierlyn Ave., Birchencliffe, Huddersfield. Tel: (0484) 31491

SPECTRUM READ. Teaches children to read. £9.95. SAE for details of our Educational programs. Bourne, 44 Haslam Crescent, Bexhill, Sussex.

Spectrum Mark Book (48K)

A full teachers mark book on computer. Holds all your classes; easily updated; automatic totals, percentages, order of merit, mean & S.D. graphs, etc. **£6.50** with manual and demo.

Mentor Software, 69 Beech Road, Halton, Lancaster.

MAGIC MESSAGE

Put **your** name in the **Hall of Fame**, be the **Hero** of the game. Find the hidden list of words in adventure games. All this is possible with **Magic Message** which lets you look at and change messages in machine code games.

Simple load **Magic Message**, play in main part of game and like magic messages will appear on your video.

POST £4.95 FREE

O/Seas, Europe + 50p, others + £1.50. (Please order as 'Magic M')

ZX GUARANTEED (ZC)
29 Chadderton Drive, Unsworth, Bury, Lancs BL9 8NL.

COMPUTER SOFTWARE BUSINESS AND GAMES FOR MOST COMPUTERS

Commodore 64, VIC 20, BBC, Atari, Dragon, Spectrum MZ700, ZX81.

New releases for CBM 64, Spectrum, BBC, Atari, Dragon, Spectrum, MZ700, ZX81.

New releases for CBM 64, Spectrum, BBC. Every 2 weeks — just send SAE and we send you a new list every 2 weeks.

Altai C10 data cassettes 40p each. 5 C15 data cassettes £2.40 5¼" floppy discs D/sided — DL density. £2.30 each or 5 for £10.

Send to: **M. J. Seaward, St. Olaf's Road, Stratton, Nr. Bude, Cornwall EX23 9AF Tel: (0288) 4179**

SERVICES

DATA DUPLICATION
MG. Copies (ZX), Burntwood, Walsall, Staffs WS7 0ES.

Data duplication, quality cassettes, printing, flexible delivery with competitive prices and fast turn around. Prices and samples on request. **0543-480887** or **05436-75375** (24 hrs)

ZX SPECTRUM
32K RAM Up-Grade, £23.50 + VAT.
Spectrum Repairs £20.00 + VAT.
Micro-Tek Electronic Services
01 - 520 6414

SOFTWARE

SPECTRUM 48. Learn Basic games programming with our fully structured cassette. All functions covered with six challenging games including Q Bert with many hazards only £5.95. Make payable to J. Schofield, 26 Bunkers Hill Road, Hattersley, Hyde, Cheshire SK14 3QA. Enquiries Welcome.

WAFA DRIVE — MICRODRIVE TRANSFER TAPE

Transfer majority of your games to **Wafa Drive** or **Microdrive** using the 007 WD or 007 MD tapes. These use the semi automatic method proved to be the **most** successful available. Includes transfering of headerless programs and programs in map area.

The Wafa Drive 007 WD includes system 2 greatly reduced loading time for multi part programs. Longer programs can be transferred to **Wafa Drive/Microdrive** using the techniiques printed in newsletters of **Microdrive/Wafa Drive Exchange**.

Scrabble, Matchpoint being two such.

007 WD or 007 MD on cassette **£5.95**
007 WD on **Wafa** **£8.95**
O/Seas, Europe + 50p others + £1.50

Microdrive/Wafadrive Exchange Club Membership is just **£6.00** for 12 issues. (**£9.00** Europe, **£12.00** elsewhere)

ZX Guaranteed (ZC), 29 Chadderton Drive, Unsworth, Bury, Lancs BL9 8NL. Tel: 061-766 4712

FOR SALE



For ZX81 & SPECTRUM · Trouble-free loading from unspoiled tapes · Switch from SAVE to LOAD without removing plugs · Passive filter on SAVE ·

· Send SAE for leaflet · · State ZX81 or SPECTRUM ·

CELINA PRODUCTS LTD (S), Daisy Walk, Upper Allen Street, Sheffield, Yorkshire S3 7PJ. Tel: 0742 756728 (Formerly Elinca Products Ltd)

£14-99
Inc. VAT – Post FREE

UTILITIES

MICRO/WAFA/DISC DRIVE TRANSFER

48k Spectrum owners with Micro/Wafa/Disc drives can NOW transfer the MAJORITY of the programs (inc. headerless, long programs, + those with LOW addresses — say 16384).

To transfer from tape you must convert the program, and then transfer the converted bytes. MD1 (or WD1) has 5 programs *giving you the tools* to do the modifications to get them running. MD1 also has two "CAT" housekeeping programs. MD1/WD1 has a comprehensive manual with example transfers and exercises.

This widely acclaimed program (see Your Spect. No 9, P14) is ESSENTIAL if you are going to transfer MOST of your programs to your drive. MD1 (WD1 for disc or Wafadrive owners) costs a mere £6.50.

***MINIDRIVE 6** — This extra program conveniently transfers Basic, Code + Arrays directly yo M/drive, saving time and trouble. Alter prog names, integrated header reader, STOPS programs, etc. etc. Only £3.

ALL PRODUCTS CARRY OUR MONET BACK GUARANTEE

Overseas: add £1 Europe, £2 others for each product. SAE for enquiries.

LERM,
DEPT ZX, 10 BRUNSWICK GARDENS, CORBY, NORTHANTS.

UTILITIES FOR THE QL

6 programs on microdrive for Sinclair QL to prevent DIRectory overflowing the screen, provide single key LOADing or DELETion of files, repeat FORMATing of cartridges and back-up copying of whole or part of the cartridge. Printing of Directory & files, also programmes toolkit. **£10** from:

W D SOFTWARE
Hill Top, St Mary, Jersey, C.I.
TEL: 0534 81392

ACCESSORIES

SPECTRUM RGB OUTPUT

Impossible? No. Our RGB OUTPUT module is the answer to your prayers.

- **True TTL level outputs plus comp syncs**
- **Can be used with any RGB/TTL monitor**
- **No modifications or internal connections**
- **Simply plugs into expansion port**
- **No separate power supply required**

Price **£34.95** + £1.00 P&P or SAE for details – also many ZX & Spectrum addons and parts.

-adapt electronics- 20 Starling Close, Buckhurst Hill, Essex. IG9 5TN

CAMBRIDGE COMPUTER DESKS

Variety of models in real wood or formica from **£45.00**.

SAE for product information.

53 Long Lane, Willingham, Cambs CB4 5LD. Tel: (0954) 60840

SUPER NEW wire and computer tidy unit to suit all computers and their hardware. At £19.99 inc. p&p and VAT. Send for details to c/o 12 The Vineries, Endfield, Middx. EN1 3DQ.

SPECTRUM SPEEDYLOAD: Halve your waiting time! This short program lets you save/load at 3000 Baud. Easy to use with any tape command. Cassette, 48K only, £2.95. Ness Micro Systems, Main St, Mlir of Ord, Ross-Shire IV6 7SU.

DON'T LEAVE IT TO CHANCE. GIVE YOUR BUSINESS A BOOST.
Advertise in ZX. Phone Jason on 01-437 0699 for details.

ACCESSORIES

Every computer needs
CHATTERBOX II
"Listen creep, I am the leader ...."



NEW! EXCLUSIVE! PITCH CONTROL

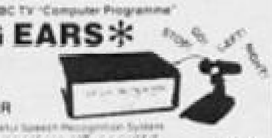
CHATTERBOX II can say anything!

Genuine phoneme synthesis - not just recorded speech - hence unlimited vocabulary. Programmable pitch for more natural intonation (exclusive to Wm Stuart Systems) - solid tone cabinet for quality sound - integral beep music amplifier. **PLUS** expansion socket for BIG EARS voice recognition system. Full instructions technical notes and software supplied with this outstanding educational unit.

DEALER ENQUIRIES WELCOME

As seen on BBC TV "Computer Programme"

BIG EARS



SPEECH INPUT FOR ANY COMPUTER

BUILT TESTED & GUARANTEED

ONLY £49

ZX81/SPECTRUM NEW! VIBRATO CONTROL

MUSIC SYNTHESISER (Stereo) + 16 LINE CONTROL PORT 16K RAM

AMAZING VALUE AT ONLY £19.50

SOFTWARE: THE COMPOSER — TALKING HANGMAN — ZX ARP/DRUMSEQ — CHROMACODE??????

COLOUR MODULATOR £16 BUILT £22

Please add VAT at 15% to prices
Barclay Access orders accepted by telephone

All enquiries S.A.E. please

WILLIAM STUART SYSTEMS Ltd

ZX COMPUTING
CLASSIFIED ADVERTISEMENT — ORDER FORM

If you have something to sell now's your chance! Don't turn the page — turn to us!

Rates of charge: 40p per word per issue (minimum of 15 words). Please state classification and post to: **ZX COMPUTING, CLASSIFIED DEPARTMENT, 1 GOLDEN SQUARE, LONDON W1.**

Please place my advert in **ZX COMPUTING** for ...... issues commencing as soon as possible.

I am enclosing my Cheque/Postal Order/International Money Order for: (delete as necessary) £..... (Made payable to A.S.P. Ltd)

OR Debit my Access/Barclaycard (Delete as necessary)

All classified advertisements must be paid for in advance.

Please use BLOCK CAPITALS and include post codes.

Name (Mr/Mrs/Miss/Ms) ....
(delete accordingly)

Address ....

....

....

Signature .... **Date** ....

Daytime Tel. No. ....

Cut out and SEND TO:

ZX COMPUTING

**ZX Computing Subscriptions,
Infonet Ltd,
Times House,
179 The Marlowes,
Hemel Hempstead,
Herts HP1 1BB,**

SUBSCRIPTION RATES

(tick ☐ as appropriate)

£13.50 for six issues ☐

£15.00 for six issues ☐

£31.00 for six issues ☐

$19.50 for six issues USA surface mail. ☐

Subscription Order Form

Please commence my subscription to ZX Computing with the very next issue.

*I am enclosing my (delete as necessary) cheque/Postal Order/International Money Order for £: ....
(made payable to ASP Ltd)
OR
Debit my Access/Barclaycard*
(*delete as necessary)*

Please use BLOCK CAPITALS and include post codes.

NAME (Mr/Mrs/Miss) ....

ADDRESS ....

....

.... **POSTCODE** ....

Signature ....

Date ....



ALIEN
In space no one can hear you scream.
NAVIGATOR
Shy, Skilful and Intelligent — Panics Easily.
EXECUTIVE OFFICER
Direct, Imaginative, Cautious, Loyal.
SCIENCE OFFICER
Secretive, Unlikeable, Brilliant — Occasionally Illogical.
CAPTAIN
Solid, Dependable, Courageous — Excellent Leader.
ENGINEERING OFFICER
Physically Strong, Low I.Q. Potentially Rebellious
3RD OFFICER
Wilful, Ambitious, Authoratitive, Resourceful.
ENGINEERING OFFICER
Cynical, Rebellious, Untrustworthy, Unflappable.
THE CREW
Personnel files follow — yours to command — well almost...
MIND GAMES
SPECTRUM 48K · CBM64
Argus Press Software Group
Featuring the unique Personality Control System
£8.99
No. 1 Golden Square, London W1R 3AB, Telephone 01-437 0626

TURBO CHARGE YOUR SPECTRUM



Outperforms any Spectrum interface

The unique Turbo interface from Ram gives you all these features – and more – in one unit:

* A variety of interfaces including: Rom cartridges, two 9-way D plugs for standard joysticks, PLUS full expansion bus at rear.
* Compatible with Kempston and Protek protocols.
* Works with latest Quickshot Mk II auto rapid-fire joysticks!
* Choice of Rom cartridge or tape cassette software.
* Instant program loading with cartridge software.
* Built-in power safety device – unique to Ram Turbo.
* Full one year guarantee.
* Immediate availability – 24 Hr despatch on receipt of P.O./credit card details (cheques – seven days).
* Incredible value – only £22.95.

So don't wait around – simply complete the coupon and send it to us today.

Or call our credit card hot line on 02514 25252. (Access and Visa welcome).

Ram Electronics (Fleet) Ltd, 106 Fleet Road, Fleet, Hampshire GU13 8PA.

24 Hr despatch for credit cards and postal orders

Please send me:

____ Spectrum Turbo Interface(s) at £22.95 + £1 p+p (overseas orders £3 p+p)

____ Quickshot II Joystick(s) at £9.95 (Only when purchased with Turbo – normally £12.95 + £1 p+p)

I enclose cheque/postal order or charge my Access/Visa for: £ ____

Name ____

Address ____

ZX

Tel ____

To: Ram Electronics (Fleet) Ltd. 106 Fleet Road, Fleet, Hampshire GU13 8PA

TURBO

Trade and export enquiries welcome.